5. SYSTEM ADMINISTRATOR REFERENCE

To assist in configuring, maintaining, and operating the AMHS, this section focuses on the areas of use, installation and configuration of a Graphical User Interface (GUI) Admin tool set; the use of Xterm UNIX script based Admin tools and several manual processes. With the conceptual information described in the previous sections and the tools and processes described here, supplemented by the site SOPs, the AMHS System Administrator should command the tools and expertise necessary to effectively support the system.

5.1 SYSTEM ADMINISTRATOR TASKS

The System Administrator monitors system activities, maintains system files, schedules and directs backups, performs restores, and oversees the operation of the primary and secondary AMHS threads on those installations that include redundancy. The Admin activities are performed in concert with the GCCS EM Server Admin tasks to ensure proper overall system functionality.

The AMHS System Administrator's duties include:

(1) Monitoring operations. The System Administrator monitors system activities in conjunction with the System Operator, periodically monitors disk utilization and the status of the network/interface, and reconfigures file systems as needed to optimize performance.*

(2) Maintaining system files. The System Administrator creates, maintains and deletes TOPIC user accounts and generates reports on user accounts, profile information and system performance.

(3) Directing backups. The System Administrator determines backup requirements and initiates backups in accordance with site SOPs and Defense Communications Agency (DCA) policies, and coordinates backup activities with the System Operator.

(4) Implementing security policies and procedures. The System Administrator performs security procedures as directed by the Security Administrator, enables and maintains the system security features needed to support security policies, and maintains the Topic password file.

(5) Controlling system configuration. The System Administrator monitors and maintains a log of all software and hardware changes to the system.

(6) SAT Operation. The System Administrator oversees the operation and maintenance of the SAT PC and is the primary interface with the TCC on AUTODIN traffic issues.

* MS Word's "Smart Quotes" cause errors in the UNIX syntax. Special care should be observed when executing commands described in this document.

(7) Redundant system operations. The System Administrator maintains a Primary and Secondary Thread, both receiving live AUTODIN traffic. (See Section 6 for further information.)

5.2 AMHS ADMINISTRATION TOOLS

The configuration and operation of the AMHS is aided by the Sys Admin Tools kit. Many of the tasks that might be done in an Xterm window or special script have been automated with GUI interfaces. These tools greatly simplify many of the Admin tasks, but special care must be exercised because the scripts manipulate the system and configuration files, which must be synchronized for the system to keep track of its state and variables. If the GUI tool set is installed, you must **only** use these tools for all functions they perform.

5.2.1 A Single Starting Point for Most Sys Admin Functions

The Sys Admin Tool GUI provides a Motif GUI similar to other GCCS EM tools, such as Security Manager and Profile Manager. With these tools the majority of the Admin functions can be performed, and site-specific existing tools can be linked into the Sys Admin through its Custom Tool Manager feature. The system retains the same GCCS system dependencies as before including the GCCS Kernel Load, Executive Manager and Command Center Applications. It still relies on existing Executive Manager Tools such as Security Manager for adding UNIX accounts, editing group access, and Profile Manager for associating users to User Profiles.

The AMHS Sys Admin Tools are launched by any user from any workstation desktop that has been identified as having the privilege to use the tool. Authorizing this privilege is a two-step process. First, the user must have the icon added to their launch list using the Profile Manager application, and second, their workstation name and username must be added to the **/h/AMHS/Server/topic/amhs_db /home/.rhosts** file. See the UNIX man **.**rhosts for more information about this feature. This file will contain a table that might look like:

alpine	**steve**
amhssvr	**steve**
sunadm	**steve**
alpine	**jane**
amhssvr	**jane**
sunadm	**jane**

The tool uses several configuration files. These files are defined in the **/h/AMHS/Server/data/config/tools/admin.ini** file as shown in Figure 5-1. The effects of editing these files will be described in Section 5.11.2.1 along with other installation and configuration information.

```
# admin.ini  (configuration file)
#
# This is the main configuration file for the
# AMHS Administration tool. This file is used to
# define the location of other configuration files
# and directories.
#
# Make sure you have a complete understanding
# of these values before making any changes.
#
MAIN_WINDOW_FILE=/h/AMHS/Server/data/config/MainWindow.ini
DAC_MANAGER_FILE=/h/AMHS/Server/data/config/DacManager.ini
QUEUE_MANAGER_FILE=/h/AMHS/Server/data/config/QueueManager.ini
CUSTOM_LAUNCH_FILE=/h/AMHS/Server/data/config/CustomLaunch.ini
SCRIPT_LIBRARY_FILE=/h/AMHS/Server/data/config/ScriptLibrary.ini
ACCOUNT_MANAGER_FILE=/h/AMHS/Server/data/config/AccountManager.ini
BACKUP_MANAGER_FILE=/h/AMHS/Server/data/config/ArchiveManager.ini

TEMP_DIRECTORY=/tmp
VARDEF_FILE=/h/AMHS/Server/topic/amhs_db/vardef
LOG_FILENAME=/h/AMHS/Server/topic/amhs_db/log/system_admin
```

Figure 5-1. Listing Of The admin.ini File

5.2.2 AMHS SA ICON MENU TREE

Launching the AMHS SA Icon opens the Main Window which displays several system status and alarm indicators. This is the launch point for the other tools. The AMHS Administration Menu Tree shows the various functions that can be activated from the Main Window.

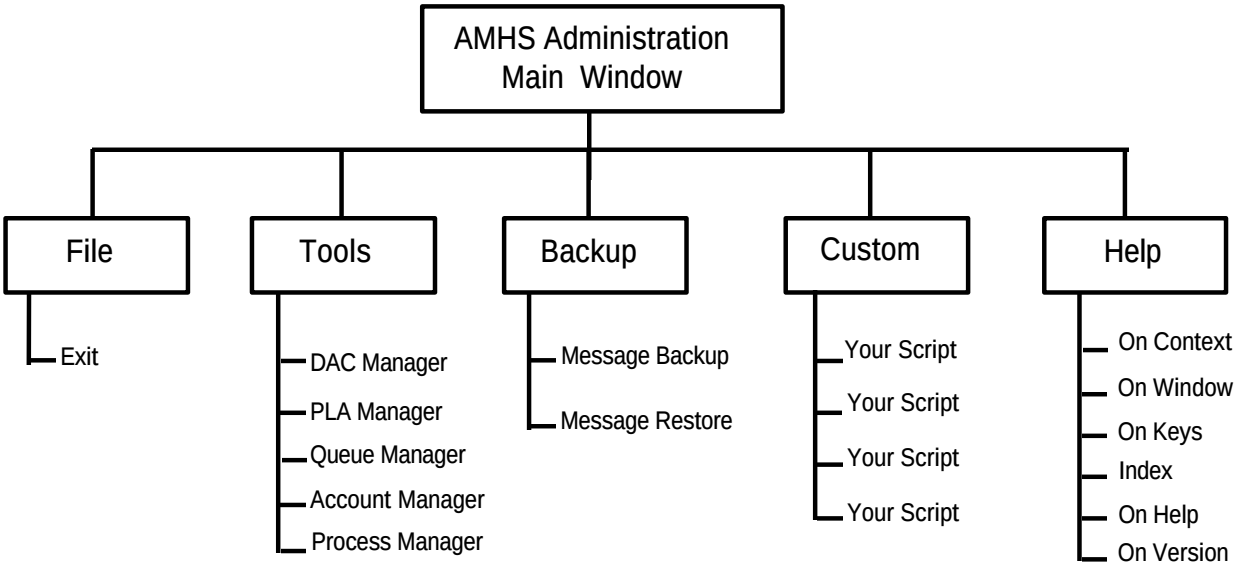


Figure 5-2. AMHS Administration Menu Tree

The Menu Tree reflects the command structure of the pull-down menus and the following sections describe their functionality. The **Tools** menu covers most of the system's management functions. **Backup** gives the operator an easy way to archive and restore messages to tape, freeing up disk storage and speeding up message browsing by reducing the number of messages on-line. This is coupled with a restore capability so archived messages can easily be located and restored on demand. Many of the administrative tasks performed in the day-to-day operation can be or are script driven. The **Custom** window is a place to keep these scripts handy and easily executed.

5.3 AMHS SA MAIN WINDOW

In the previous section, Figure 5-2 described the menu bar pull-downs. In Figure 5-3, the Main Window displays several typical status lines that are identified in the **MainWindow.ini** default file. The first two lines display the time the last AUTODIN messages were received and transmitted. These fields are not configurable.

The AMHS Server PROCESSOR STATUS indicates the server and selected processes are up and running. Other servers can be added to the list. The processes monitored are via the EM Monitor and Control APIs. The concepts and operation of the Monitor and Control EM subsystem are defined in Appendix C, Section C.4.

The QUEUE STATUS indicates how many messages (files) are stacked up in each Queue (subdirectory). The adjacent OK/ALARM indicators each have adjustable thresholds to alert the operators when the queues are filling up, which indicates either the system has slowed down or some process is not running properly. Adding a queue for the **pf1** mailbox would be useful to indicate a need to split the profilers into **pf1** and **pf2**, for example.

The DISK STATUS indicates the percentage of hard drive space that has been filled and can be used for planning message backups or log purges.

When the **pf1** mailbox starts overflowing, regularly it indicates that the **pf1** processes cannot keep up with the message traffic level. Splitting the process into **pf1** and **pf2** allows the system to share the load between two processes.

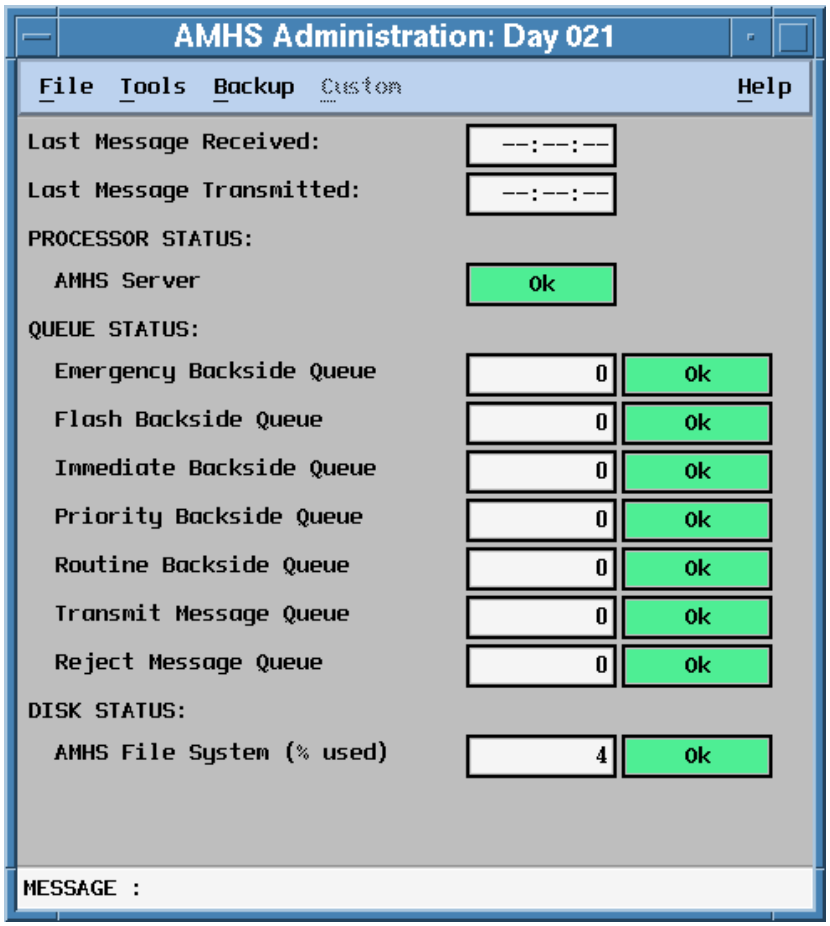


Figure 5-3. AMHS Administration Tool

5.3.1 SA Main Window

This section describes the main window display. The main window display is divided into four distinct sections that provide different status information. This section will explain the functionality that each section provides and also demonstrates how this main window can be configured to suit your site's requirements. The four sections of the main window are divided into a message transmission, a processor status, a queue status, and a disk status area. Figure 5.3 displays a typical main window.

Each status area of the main window updates in real-time. In other words, there is no need to refresh the display in order to see the current status for each section. As delivered, the update period is set to 60 seconds. Refer to the section on the **MainWindow.ini** file to change this time period. For performance reasons, it is recommended to use a period that is no shorter than 60 seconds. You may also wish to set this period high, to a value of 240 seconds, for example, if the real-time feature intervention becomes annoying.

With the exception of specifying your processor hostname for status, the contents of the **MainWindow.ini** file (Section 5.11.2.3) were carefully chosen to reflect the items that an AMHS system should monitor. Careful thought should go into the decision to add or delete items from the configuration file. Keep in mind when making changes—the more items that are monitored, the slower the tool becomes. The update period must be increased as more items are monitored to compensate for these delays.

The message transmission status occupies the first two lines on the main window dialog. These lines indicate the last message that was transmitted and received into the system via the SAT for the current Julian day. The current Julian day is displayed as part of the main window title bar. The times are indicated in hours:minutes:seconds format. A time of --:--:-- is displayed if no messages are detected for the current Julian day. The times are determined by analyzing the current day's SAT transmit and receive archives. The time of the last detected message is passed back to the tool and displayed in the appropriate status box.

The processor status section displays the current status of the configured server(s). Each processor status, there can be more than one, is indicated by a colored button with text appearing in the center of the button. The colored button will be any of four possible colors (green, yellow, red, or orange). The text could be (OK, Degraded, Down, or Broke). Each status reflects the following:

Text	Color	Description
OK	Green	Respective processor is normal (this is good).
Degraded	Yellow	One or more of the monitored processes have stopped.
Down	Red	Unable to obtain status. The processor is not responding to monitor and control messages.
Broke	Orange	Your MainWindow.ini file contains an invalid processor name.

The processor status is gathered using the Monitor and Control (M&C) APIs that are part of the Executive Manager (EM) segment. If problems occur with obtaining processor status, it is most likely a problem with the installation of your EM segment. It is possible to add more servers to this section of the main window by editing the contents of the Processor Status section of the **MainWindow.ini** configuration file. For example, if your site has a redundant AMHS Server, you may wish to add that processor to the list of monitored processors. Only processors that have the EM segment installed can be monitored.

Each processor is monitored down to the processes that are running on the workstation. Not all processes on the workstation are monitored. Only certain designated processes for AMHS are monitored. These processes are specified in file **/h/EM/config/active_spt** listed in Section 5.11.4. Refer to the EM description document on M&C for an explanation of this file. The AMHS processes are usually included in this file when your server was installed. In other words, it is part of the initial AMHS installation.

The queue status portion of the main window contains one line of information for each configured queue. The status of each queue is communicated via two display boxes. One box, count status box, contains a number that represents the number of entries (files) that are in a queue (directory). The second box, colored status box, is a colored button that indicates whether the queue has reached its configured threshold. Each queue is defined in the **MainWindow.ini** configuration file. The configuration of each queue includes specifying a queue name, queue location (directory name), and a threshold. The colored status box can be one of two colors. The box will either be green or red. A green status indicates that the number of entries in the queue has not reached the configured threshold. A status of red indicates that the number of entries in the queue exceeds the configured threshold. Once again, refer to the section on the **MainWindow.ini** file for more information on how to configure additional queues.

The queue status information is obtained using an internal counting mechanism that simply counts the number of files within a specified directory. The number is translated into the information that appears in the count status box. The count status box contents are compared with the configured threshold to determine the color status box.

The disk status portion of the main window contains one line of information for each configured file system that requires monitoring. The status information is communicated via two status boxes—the percentage status box and the color status box. The percentage status box contains the percentage of disk space used on the specified file system (this is always reported as a percentage). The color status box is used to determine if the percentage used exceeds the configured threshold. File systems that are monitored are contained in the **MainWindow.ini** file.

The disk status is calculated by obtaining file system statistics directly from the UNIX kernel. These statistics are used to calculate the percentage of disk space used. The calculated percentage is translated into a string that appears in the percentage status box. The percentage is then compared with the configured percentage threshold to obtain the color of the color status box.

5.3.2 System Status Monitoring

This file lists the processes monitored, and the availability of each is typically "AND-ed" to display PROCESSOR STATUS. If any of the resources degrade, the alarm will change color to indicate how significant the fault is. This indicator will help in troubleshooting system problems by giving near real-time status. The list of processes is found in **/h/EM/config/active_spt**, as shown in the following listing.

```
# active_spt
#
# This is the active configuration file for the System
# Process Table. The system process table contains a list of process
# names and host ids which represent the set of processes that the
# Monitor and Control System actively monitors. This file is originally
# based on the process_table file. The process_table will act as the
# master table and this file will be the active table.
```

(The complete file listing is in Section 5.11.4.)

```
#-------------------------------------------------------------------------
# AMHS Server Segment: List of Processes for the Server to Monitor
#-------------------------------------------------------------------------
satfeed1#sun3#SAT Feed#sat_feed
cbcfeed1#sun3#CBC Feed#cbc_feed
rtmsgqpr#sun3#Queue Profiler#rt prof -PROCNAME pf0
rtmsgpro#sun3#Message Profiler#rt prof -PROCNAME pf1
rtdbserv#sun3#Database Server#rt server -PROCNAME server
cbcmerge#sun3#CBC Message Merger#rt merge -PROCNAME mg4
autodinm#sun3#AUTODIN Message Merger#rt merge -PROCNAME mg1
cbcdatap#sun3#CBC Data Preparation#rt build -PROCNAME dp4
autodind#sun3#AUTODIN Data Preparation#rt build -PROCNAME dp1
#-------------------------------------------------------------------------
```

5.3.3 Sys Admin Tools Main Window Operating Instructions

The menu bar is the launch point for each of the suboperations for the tools. See Figure 5-4.

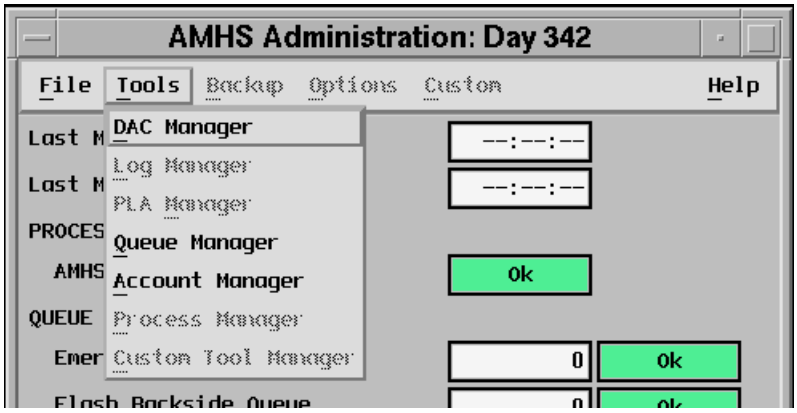


Figure 5-4. SA Main Window Pull-Down Options

5.3.4 Adding Additional Queue Status Monitoring

One of the possible bottlenecks for system performance is the ratio between users, profiles and profilers (**pfx**). The number of messages backed up in each topic **pfx** mailbox is a good indicator of this. One solution is to monitor these mailboxes and use the resulting information to

assist in planning how and when to modify the system configuration. Setting up this monitoring function is as simple as adding one more entry to the **MainWindow.ini** file.

QUEUE_ENTRY=Routine Backside Queue,/h/AMHS/Server/sat/autodin/bsq5,10
QUEUE_ENTRY=Transmit Message Queue,/h/AMHS/Server/sat/autodin/xmit,2
QUEUE_ENTRY=Reject Message Queue,/h/AMHS/Server/sat/autodin/reject,2
QUEUE_ENTRY=Profiler1Mailbox, /h/AMHS/topic/amhs_db/mailbox/pf1/msg,20

With this entry another Queue Status line and indicator will be added that will monitor the pf1 mailbox directory and flag an alarm when 20 or more messages are backed-up.

5.4 DAC MANAGER

DAC Manager Manages DAC types (Add, Change and Delete), allows code word assignment to DAC types and allows UNIX group and permission assignment to DAC types. This window, in concert with the Account Manager window, easily connects users to message profiles and security classification. See Figure 5-5.

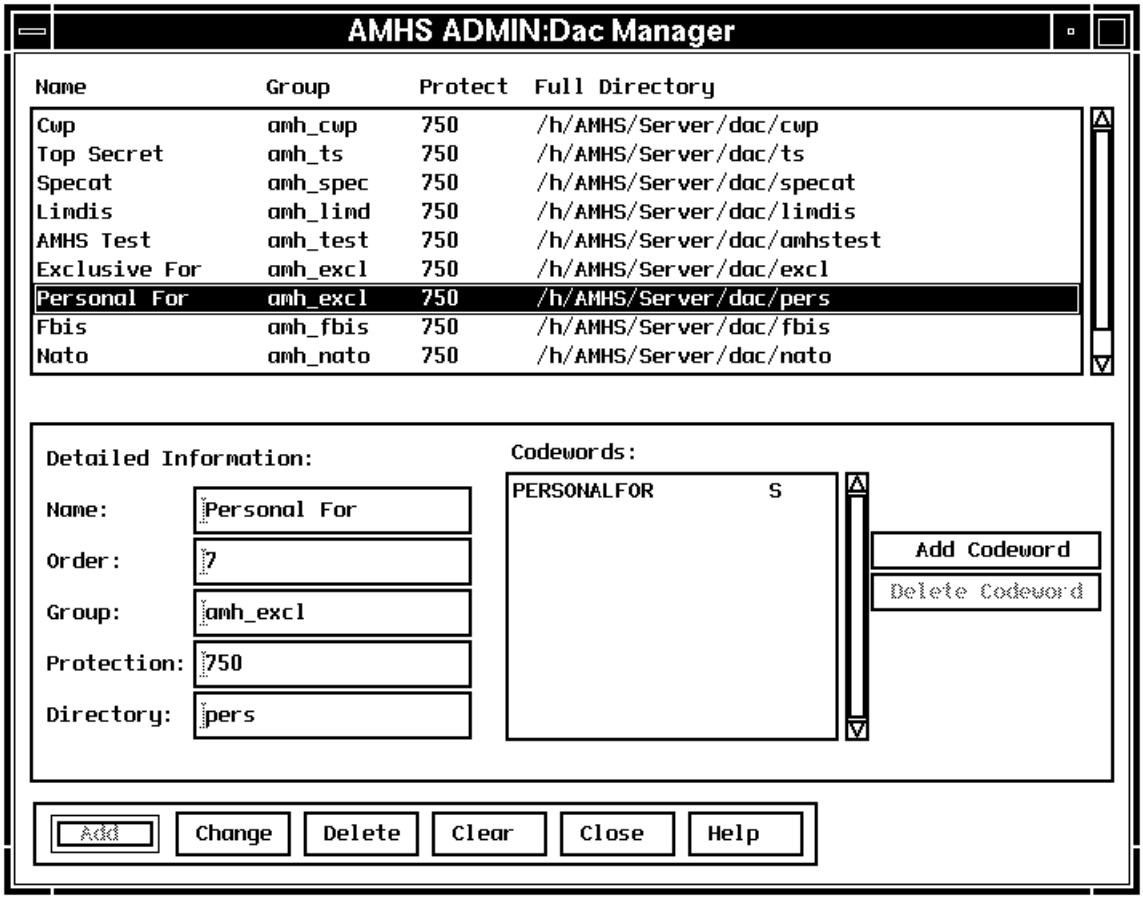


Figure 5-5. DAC Manager Window

5.4.1 DAC Manager Operation

This section describes the function and operation of the DAC manager tool. The DAC manager tool is used to manage the contents of the **daclist** file. The **daclist** file is located in the following file:

/h/AMHS/Server/topic/amhs_db/daclist

The **daclist** file is used to describe the discretionary access control groups that are part of the AMHS. The contents of this file are managed by each site as determined by the requirements for message grouping and protection. Refer to the contents of the **daclist**, itself, for a more detailed description of what the **daclist** can do. Also, refer to the appropriate section of this document for further explanation of the **daclist** file.

The DAC manager tool is used to handle all aspects of editing the **daclist** file. Everything from adding, deleting, and modifying **daclist** entries can be performed from this tool. A typical DAC manager session is displayed in Figure 5-5. This tool serves as a replacement to editing the **daclist** using vi or some other UNIX editor (see Figure 5-6).

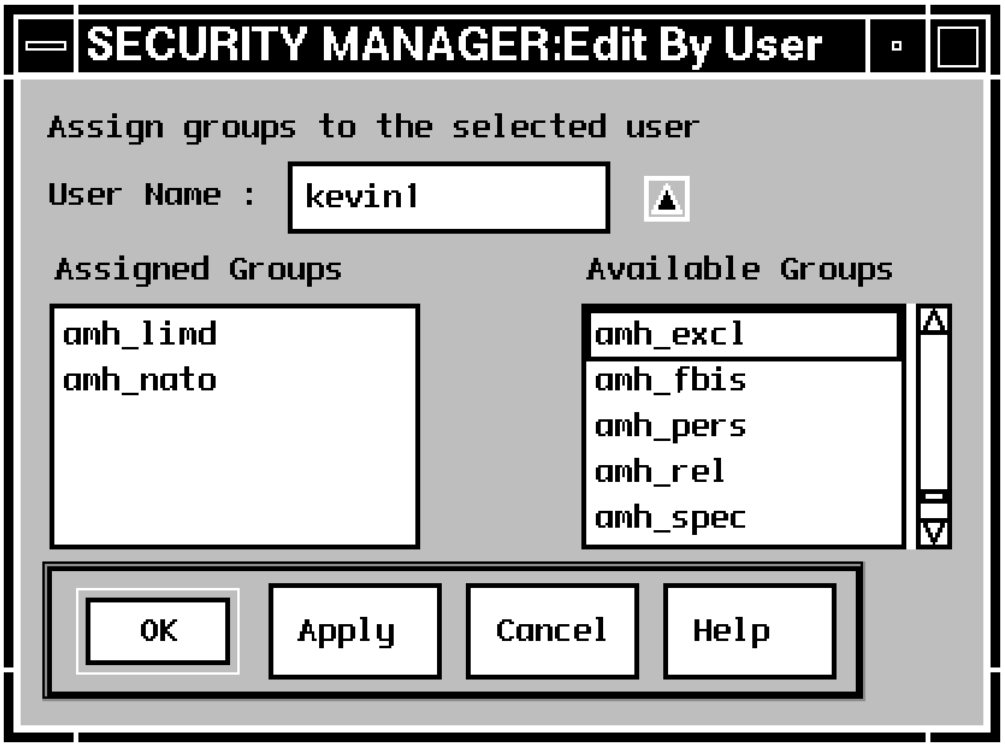


Figure 5-6. UNIX's Group Editor

The DAC manager dialog window can be divided into three sections. These sections consist of a DAC entry display list (top portion), a detailed DAC entry display (middle portion), and the action buttons (lower portion). The DAC entries display list contains a list of all the current DAC entries in precedence order. In other words, higher precedence DAC entries are at the top of the list. The name, UNIX group, UNIX file system permission, and directory name for each DAC entry is display. The detailed DAC entry display (middle portion of the dialog window) area shows complete information for a selected DAC entry. In addition to the information presented in the summary list, the order and code words that make up the entry are displayed. The action buttons section (bottom portion) of the dialog window contains the push-buttons that are used to make changes to the **daclist**.

Keep in mind that this tool is simply a front end GUI editor for the **daclist** file. This tool has no direct interaction with **sat_feed** and **cbc_feed** processes that read the **daclist**. Permanent changes to the **daclist** file will not be communicated to the feed processes. Feed processes must be restarted for **daclist** changes to take effect. Figure 5-7 illustrates the relationship between the DAC manager tool, the **daclist** file, and the feed processes. Also, the DAC manager tool does not automatically create the UNIX groups that are associated with each DAC entry. It is up to the administrators to ensure that a UNIX group exists for each DAC entry (see Figure 5-6). A feed process will not start if a DAC entry does not have an existing UNIX group. An exit message will appear in the feed's log file when the feed attempts to read the **daclist** file.

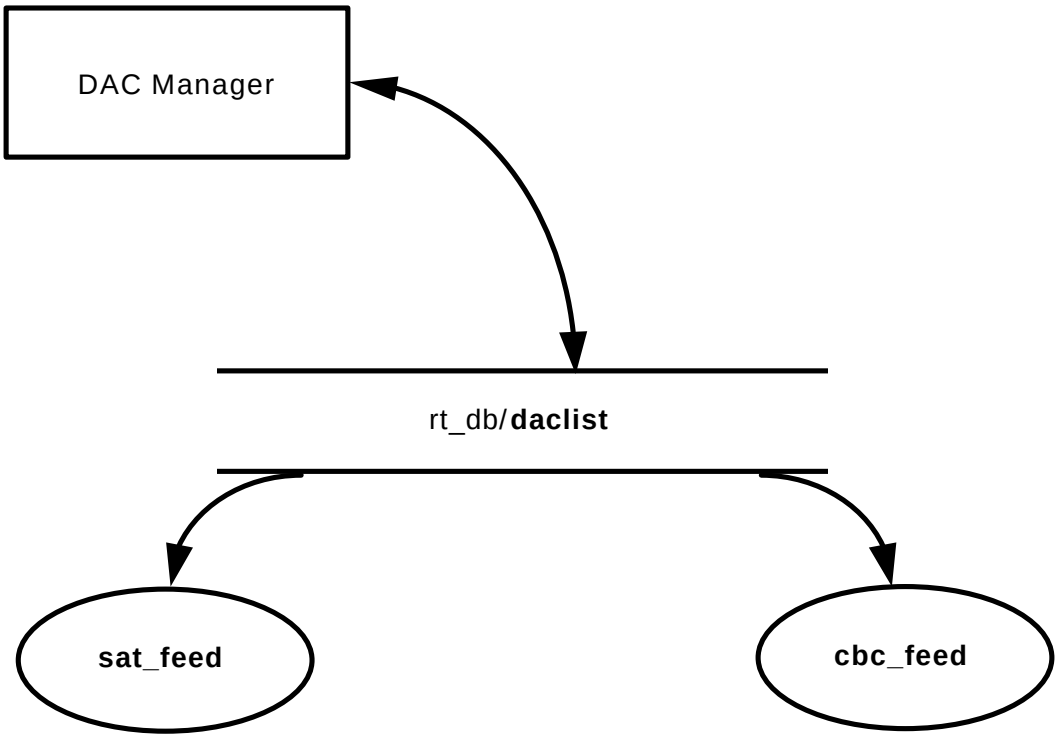


Figure 5-7. DAC Process

Incoming messages that fail to match any of the specified code words will fall into a general message category. These messages will have an owner of **amhs_dba** and the group of **gccs**. When adding new Topic user accounts, it is recommended to make all users members of this general message group. Refer to the Account Manager section for a description of security values. Also, remember there are only 15 possible DAC entries. TOPIC restricts the number of groups to 16. Think before adding a new DAC entry. Ask yourself if the DAC entry is really necessary and if you can accomplish the same task by simply adding more code words to an existing DAC entry. See Figure 5-8.

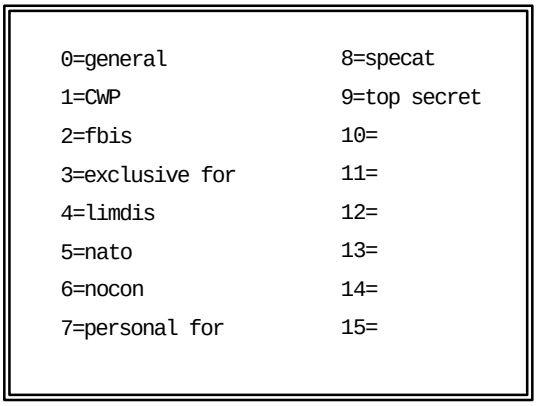


Figure 5-8. DAC Groups

daclist Restrictions

The **daclist** file uses four delimiters that may NOT be used in any of the above **daclist** definitions. They are colons, semi-colons, equal signs and “at” signs (:;=@). Any attempt to enter these characters in the fields will result in a feed error message.

5.5 PLA MANAGER

PLA Manager is not implemented in Sys Admin Tools Version 1.3. The exact description of all its features is still being evaluated. The main features at this point are: 1) View PLA Table, 2) Edit / Add PLA Entry, 3) Delete PLA Entry. (See Figures 5-9 and 5-10.) A button will allow for easy recompiles on command.

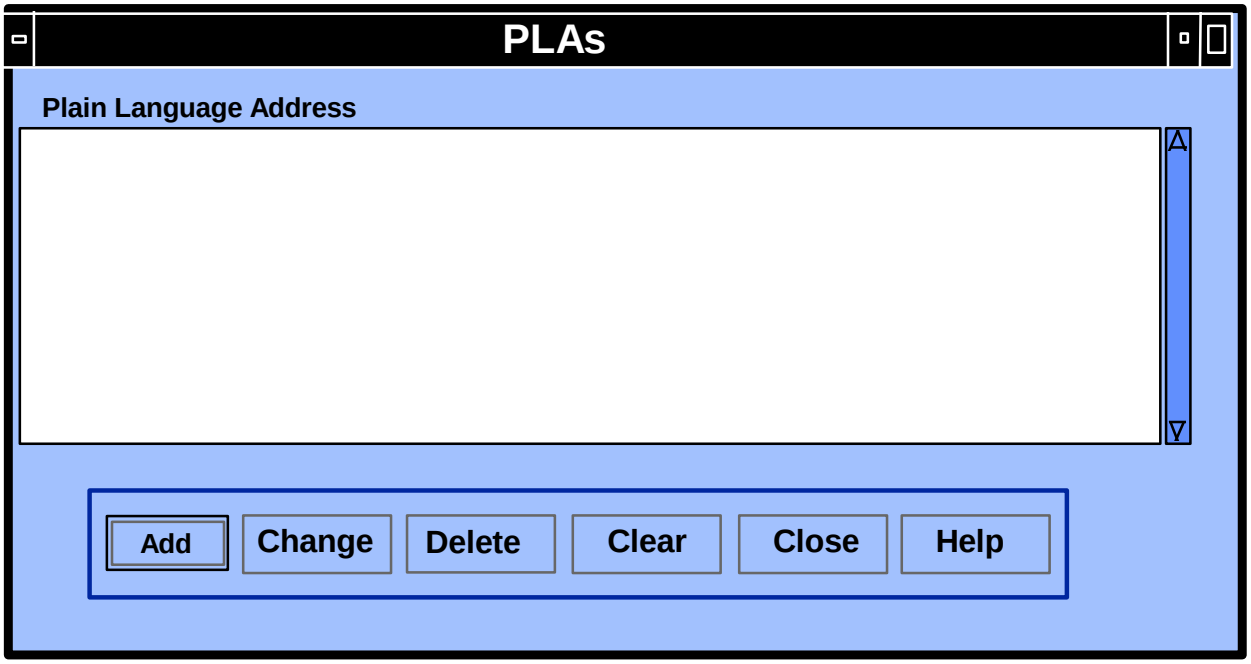


Figure 5-9. PLA Manager

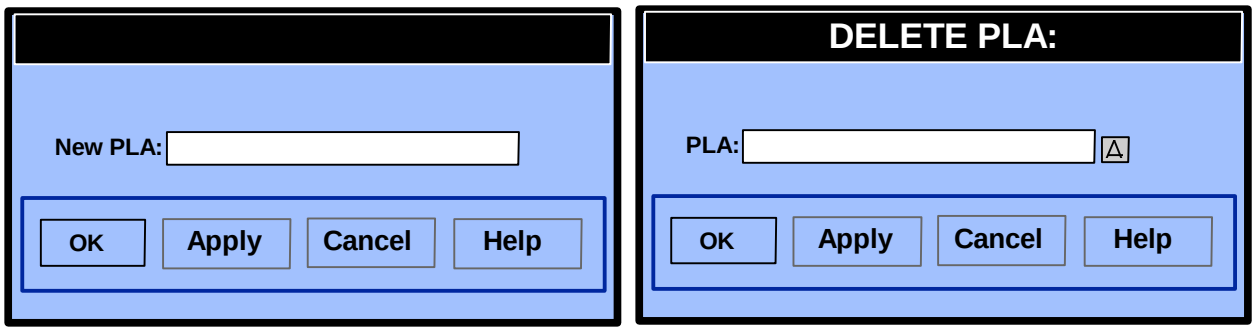


Figure 5-10. PLA Options

5.6 QUEUE MANAGER

Queue Manager is used to manage the Topic Profiles file. This tool is used to assign search criteria (topics) for message delivery. It allows Administrators to assign topics to a user's Info, Action, and Comeback queues. See Figure 5-11.

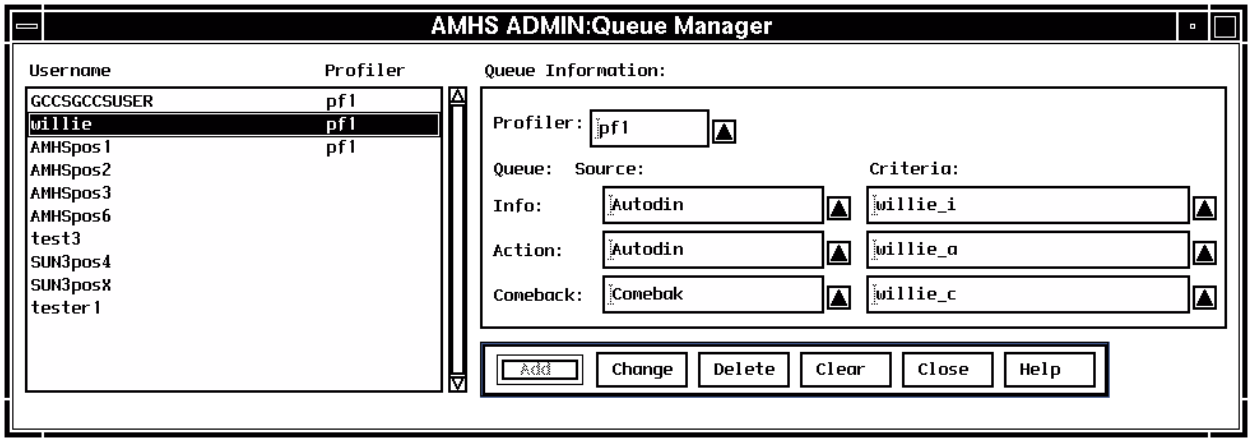


Figure 5-11. Queue Manager

5.6.1 Queue Manager Operation

The Queue Manager tool is used to define a user's message queue. Defining a user's message queue is what makes the AMHS so powerful. By carefully designing the queue, a user can be configured to only receive messages they are interested in receiving. Each user is allowed three queues that can be configured. These are the Action, Info, and Comeback queues. The Queue Manager is a replacement to editing Topic profiles and outline files. Once this tool is configured and operational, administrators should not edit the Topic profiles manually unless instructed by technical support representative. Editing these files could cause the tool to malfunction.

The Queue Manager dialog window contains a queue summary list on the left. This summary list contains a list of each Topic user. Along with each Topic user is the name of the profiler that has been configured to handle the respective user's queue. An absent profiler name indicates that the respective is not assigned to a profiler. In other words, his queues are not defined. New entries are added to the summary whenever a Topic user is added (see section on the Account Manager). The right portion of the dialog window displays the detailed information about a selected user's queues. The information includes the profiler name, the document source and search criteria for each queue. The list of available search criteria is retrieved from the Topic query database each time the criteria selection list is posted. Typically, administrators will create topics using the Topic query manager and assign those topics using this tool. Only root level topics are displayed in the criteria selection list. Figure 5-11 depicts a typical Queue Manager session. This display indicates three users that have their message queue handled by the **pf1** profiler process. For the selected user, "willie", all three queues have been defined with the following information:

Queue	Source	Criteria
Info	Autodin	willie-i
Action	Autodin	willie-a
Comeback	Comeback	willie-c

The search criteria (i.e., willie-i) was created using the Topic Query Editor .

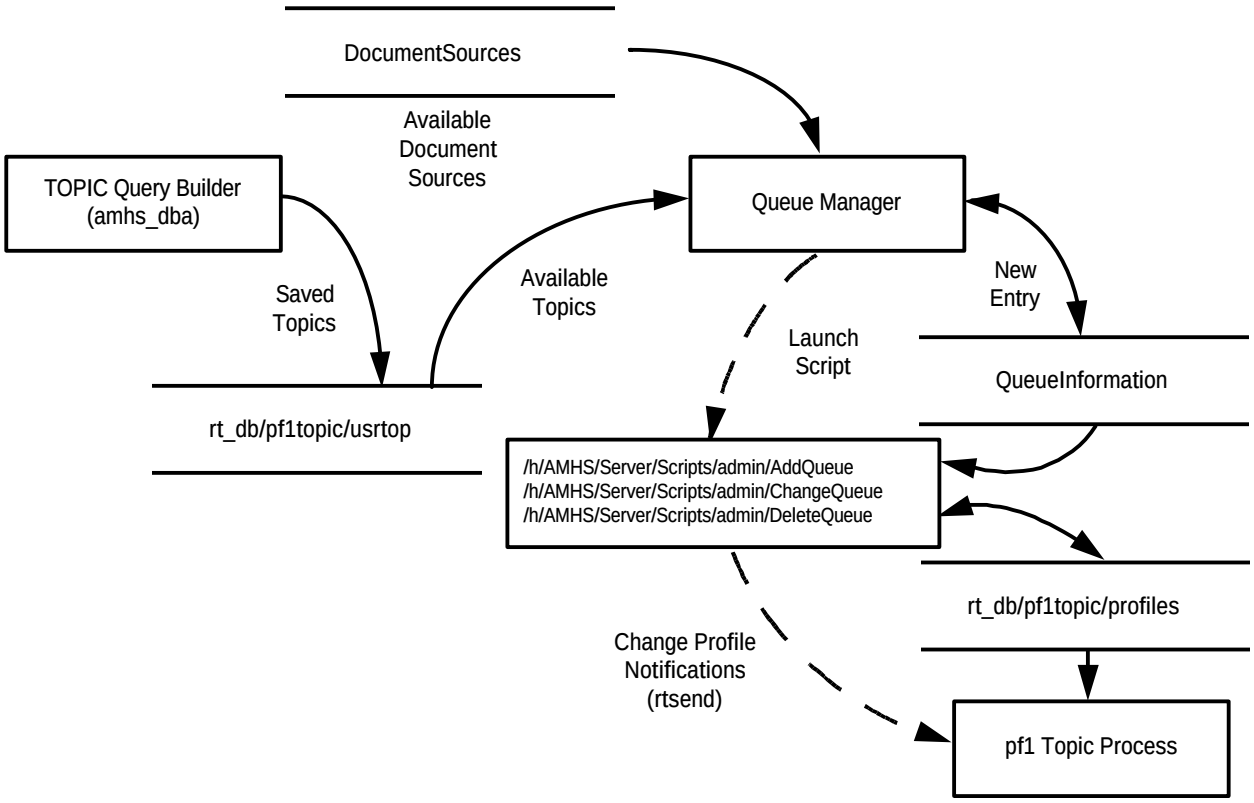


Figure 5-12. Queue Manager Processes

The Queue Manager application works by maintaining a bridge (intermediate) file that reflects the current state of all the message queues. When changes are made to the bridge file, a script is executed that makes the corresponding changes to the Topic profiles files, and the Topic profiler processes are signaled to reread their profiles files. Figure 5-12 illustrates the basic operational flow of this tool. The bridge file for this tool is called:

/h/AMHS/Server/data/admin/QueueInformation

Any changes made to the queues are recorded in this file. Whenever changes are made to the QueueInformation file, a backup copy of the previous contents is made in the following file:

/h/AMHS/Server/data/admin/QueueInformation.previous

Whenever you suspect an error has occurred using this tool, you should immediately make a copy of the backup file to ensure 100% recoverability. A backup copy can be made by executing a simple UNIX copy command from an Xterm window. A backup copy of the **QueueInformation** will ensure that your Topic profiles file can be reconstructed. Refer to the section on configuring the tool for an explanation on creating Topic profiles from a **QueueInformation** file.

5.6.2 Queue Manager And Topic Editor

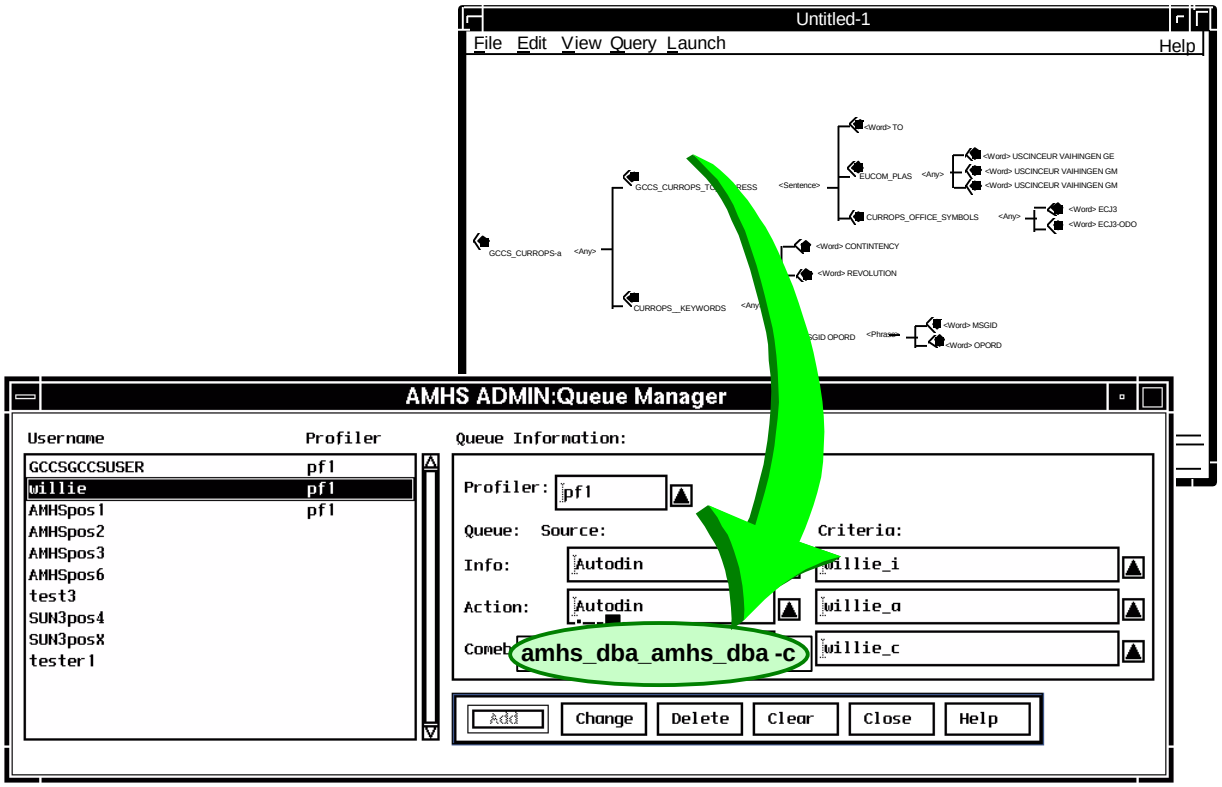


Figure 5-13. Queue Manager And Topic Editor

5.7 ACCOUNT MANAGER

The Account Manager administration tool is used to manage AMHS accounts, e.g., add, delete and changes. It also displays association to GCCS login accounts, allows assignment of DAC groups. See Figure 5-14.

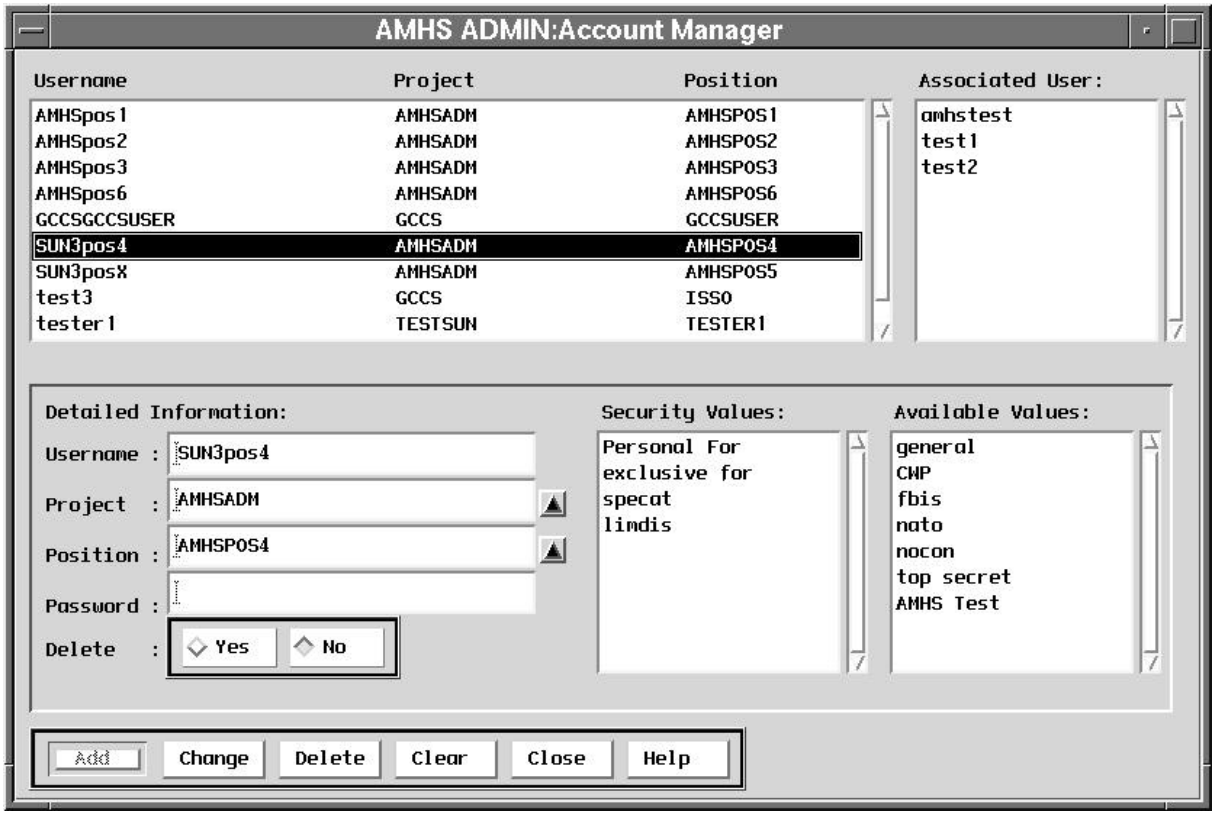


Figure 5-14. Account Manager

5.7.1 Account Manager Operation

The Account Manager tool is used to manage Topic user accounts. This tool is used to add, change, and delete Topic user accounts. The Account Manager dialog window consists of three sections. The top portion displays a summary of all the current Topic accounts. The middle portion displays detailed information about a selected account. The bottom portion contains the action buttons that are used to perform an add, change, or delete operation. It is not recommended to manually edit the Topic password file once this tool is configured and operational. It is possible to render the tool useless if ill-formatted changes are made to the Topic password file. Figure 5-14 display a typical Account Manager dialog window.

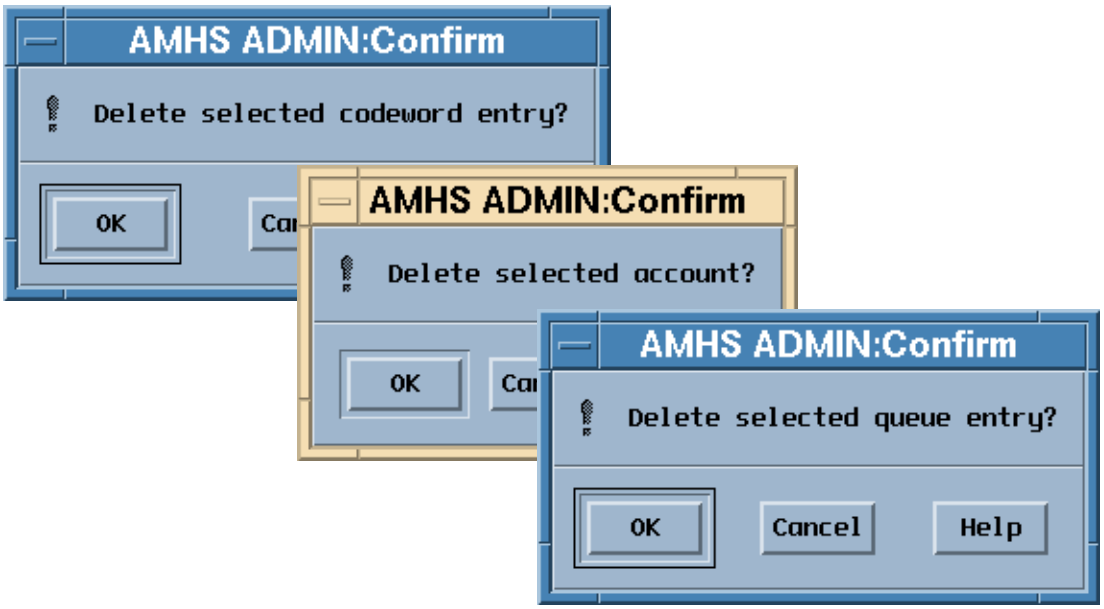


Figure 5-15. Confirmation Windows

The top portion, or summary list portion, displays all the current Topic accounts by account name. Along with each account name is the associated GCCS project and position for the account. Entries with N/A in these fields are personal accounts and are not associated with a GCCS project/position. Selecting a user from the summary list will retrieve information that is displayed in the detailed portion of the dialog window. Also, when a user is selected, a list of all the GCCS user accounts that can access the Topic account are displayed in the associated user list to the right of the summary list.

The middle or detailed information portion of the dialog window contains account attributes. These include the optional Topic password, delete permission, and security values. Accounts with a Topic password must specify the password whenever they wish to gain access to the AMHS database. Accounts with delete permission are able to delete messages from the Topic database. The security values for a user determine which messages that user can access.

All security groups except the general message group are created and destroyed using the DAC Manager tool. Users that are added to security groups must be members of the corresponding UNIX group in order to access messages that are protected by the UNIX group. Refer to the explanation on the Security Manager application for specific details. It is recommended to grant all users access to the general security group.

5.7.2 Account Manager Description

The Account Manager tool works by managing a bridge (intermediate) file that contains Topic account information. Changes made using the Account Manager tool are reflected in the bridge file. Once the changes are made, a script is launched that updates the Topic password file and notifies the Topic server process that a change has been made. Figure 5-16 shows the relationship between the Account Manager tool, the bridge file, Topic password file, and the Topic server process. The bridge file for this tool is called:

/h/AMHS/Server/data/admin/UserAccountList

A backup copy of the bridge file is made subsequent to any operation. If you suspect an error when using this tool, stop working and make a permanent backup of the previous bridge file. A backup will ensure 100% recovery. The previous bridge file is called:

/h/AMHS/Server/data/admin/UserAccountList.previous

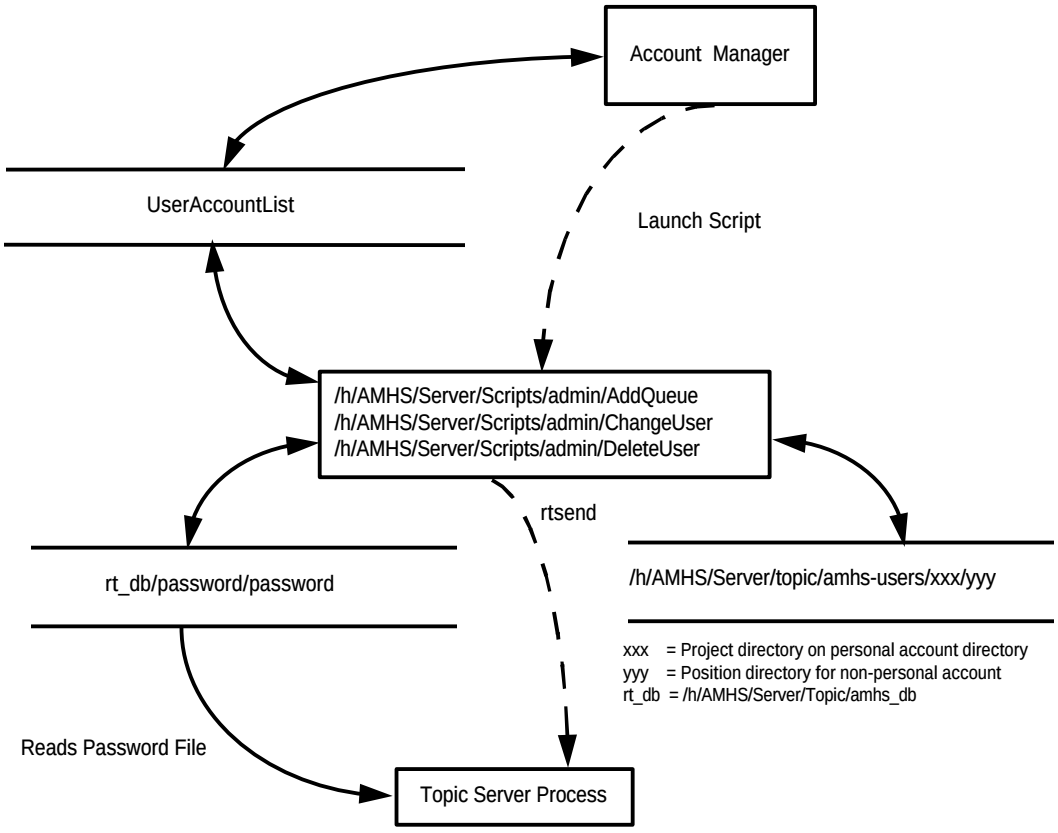


Figure 5-16. Account Manager Processes

Refer to the section on configuring the tool for an explanation on creating a new Topic password file from a **UserAccountList** file.

5.8 PROCESS MANAGER

Process Manager is the primary tool used for process level monitoring. Color coded alarm lights allow easy recognition of problems, and detailed process information is made available for each monitored process. It uses GCCS EM Monitor and Control APIs for monitoring, allowing selective process control including startup and shutdown. The process status updates in real-time. See Figure 5-17.

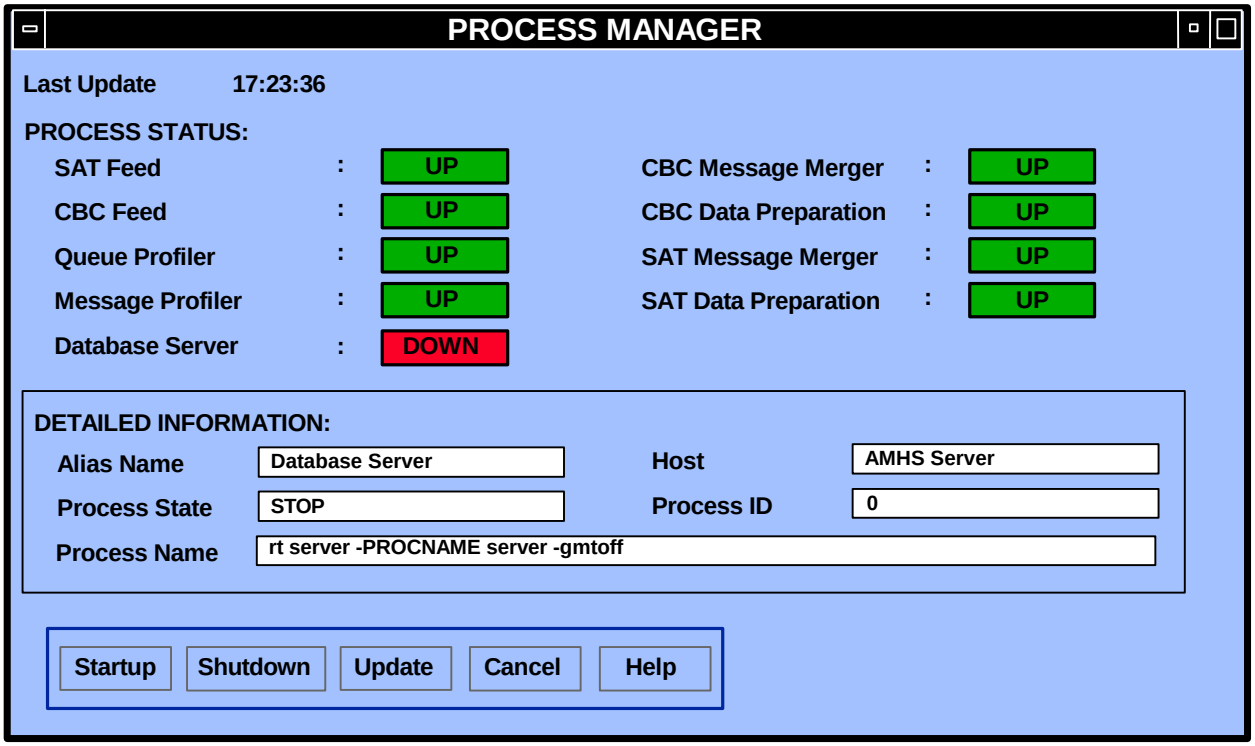


Figure 5-17. Process Manager

5.8.1 Process Manager Operation

This tool is used to manage the AMHS processes. Each configured process can be started or shut down using this tool. The functions of this tool are exactly the same as the well-known **topic_cmd** that can also be used for process control. Whereas the main window using **topic_cmd** displayed an overall status of a configured AMHS Server, this window displays a process-by-process status of the AMHS Server. Along with each process are the UNIX process ID, process name, process state, and host server name. Figure 5-17 illustrates a typical Process Manager dialog window.

5.8.2 Process Manager Description

The Process Manager will control processes similar to those shown in Figure 5-18.

```
#-------------------------------------------------------------------------
satfeed1#sun3#SAT Feed#sat_feed
cbcfeed1#sun3#CBC Feed#cbc_feed
rtmsgqpr#sun3#Queue Profiler#rt prof -PROCNAME pf0
rtmsgpro#sun3#Message Profiler#rt prof -PROCNAME pf1
rtdbserv#sun3#Database Server#rt server -PROCNAME server
cbcmerge#sun3#CBC Message Merger#rt merge -PROCNAME mg4
autodinm#sun3#AUTODIN Message Merger#rt merge -PROCNAME mg1
cbcdatap#sun3#CBC Data Preparation#rt build -PROCNAME dp4
autodind#sun3#AUTODIN Data Preparation#rt build -PROCNAME dp1
#-------------------------------------------------------------------------
```

Figure 5-18. Process Manager

As with the main window process status, this tool relies on the EM APIs for process status and control. If there are problems with status and control of processes, there is most likely a problem with the EM portion of your installation.

5.9 MESSAGE BACKUP

Message Backup (see Figure 5-19) allows administrators to select files based on level, period, and type. The level can be Daily, Weekly or Monthly. The message backup period specifies the starting day of the backup. The level specifies the number of days from the start date, e.g. daily = 1, weekly = 7, monthly = 30. The type can be AUTODIN Comeback or other sources (e.g., Reuters, UPI). Message Backup maintains a visible log of previous backup sets. A table of contents is stored with every archive on the tape header. It allows a “per message” restore capability to delete backup sets after archive is complete.

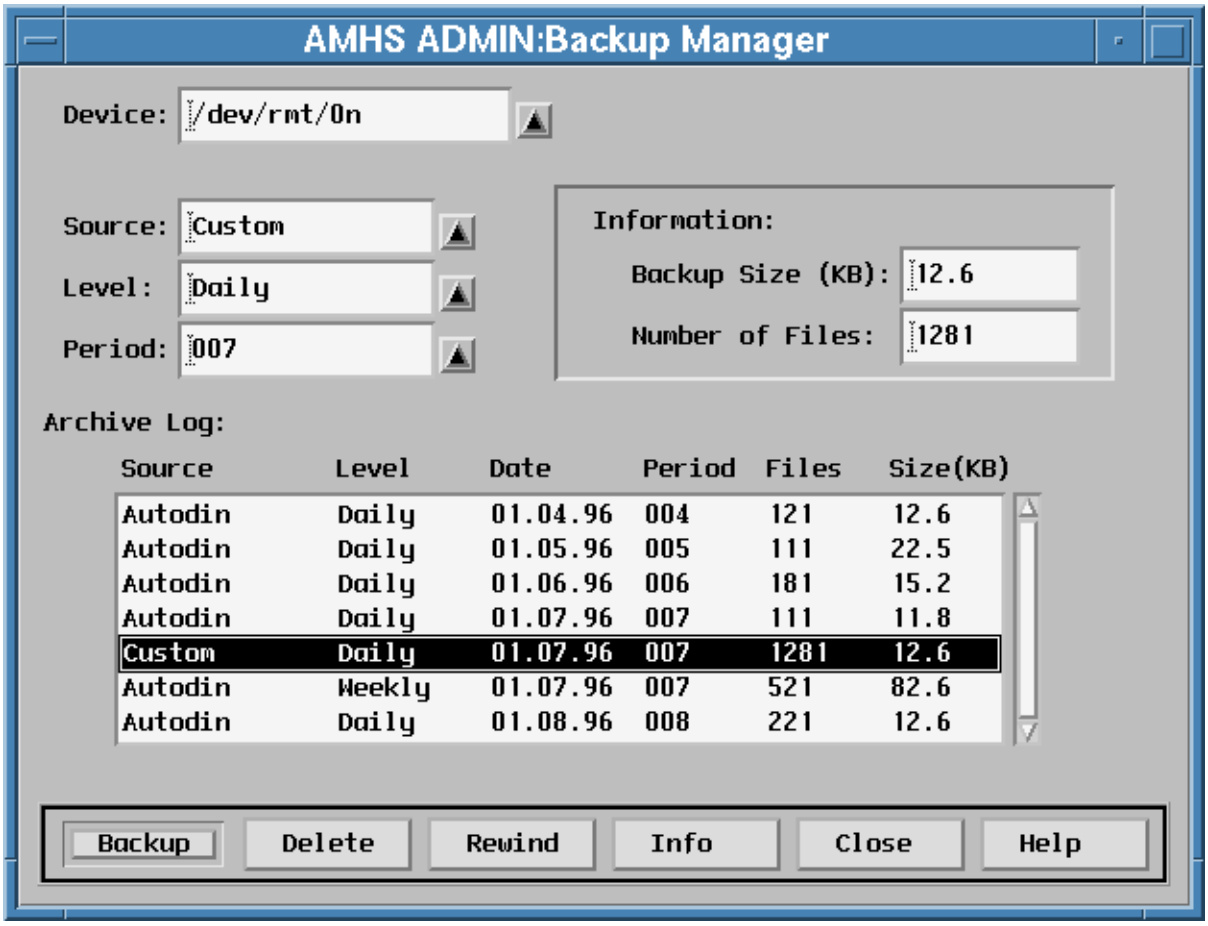


Figure 5-19. Message Backup

5.9.1 Message Backup Operation

The Backup Manager tool is used to archive and delete messages from your AMHS Server. This is not a replacement for normal UNIX system backups. This is simply a convenient means for archiving and removing message files from your servers. During the course of normal AMHS operation, the disks on the server will begin to fill. In accordance with on-line message requirements, this information needs to be removed in order to ensure disk space is available when new messages are received.

The Backup Manager dialog window contains an archive log. This log is simply a history of the information that has been archived. It should serve the purpose of quickly answering questions such as: Did I backup the AUTODIN archive on 1/2/96? The top portion of the dialog window is used to select the criteria that determines what information is archived. The current information fields will give you an estimate on the size of the backup and the number of files that are involved. The current information fields are updated whenever the “Info” button is pressed. Figure 5-19 show a typical Backup Manager dialog window.

5.9.2 Message Backup Description

TBS

Figure 5-20 Message Backup Processes

The archive consists of a series of tape archive, or “tar” images. There will be a total of (2 x N) + 1 images per archive (where N is the number of days). In other words, a seven-day archive will contain 15 tar images. This bit of information becomes important if you wish to include more than one archive per tape. The first image on the tape is the table of contents, followed by N file listings, followed by N images of the archive data.

It is possible to configure customized archive types for archiving such items as message logs and audit information. This configuration process involves editing several files that control the basic operations of this tool. It is also necessary to construct a set of backup scripts for each configured type to ensure that archives adhere to the (2 x N) + 1 rule.

5.10 MESSAGE RESTORE

Message restore allows the administrator to restore archive sets created by the backup tool. Its filters provide a way of finding a particular message by DTG. Also, archived messages can be viewed, refed, or restored back onto the server and will be processed again as if they had just come from the SAT. Full restoration of Topic partitions is done only on a complete day restore. This tool is used to restore archives that were archived using the Backup Manager. Figure 5-21 shows a typical Restore Manager dialog window.

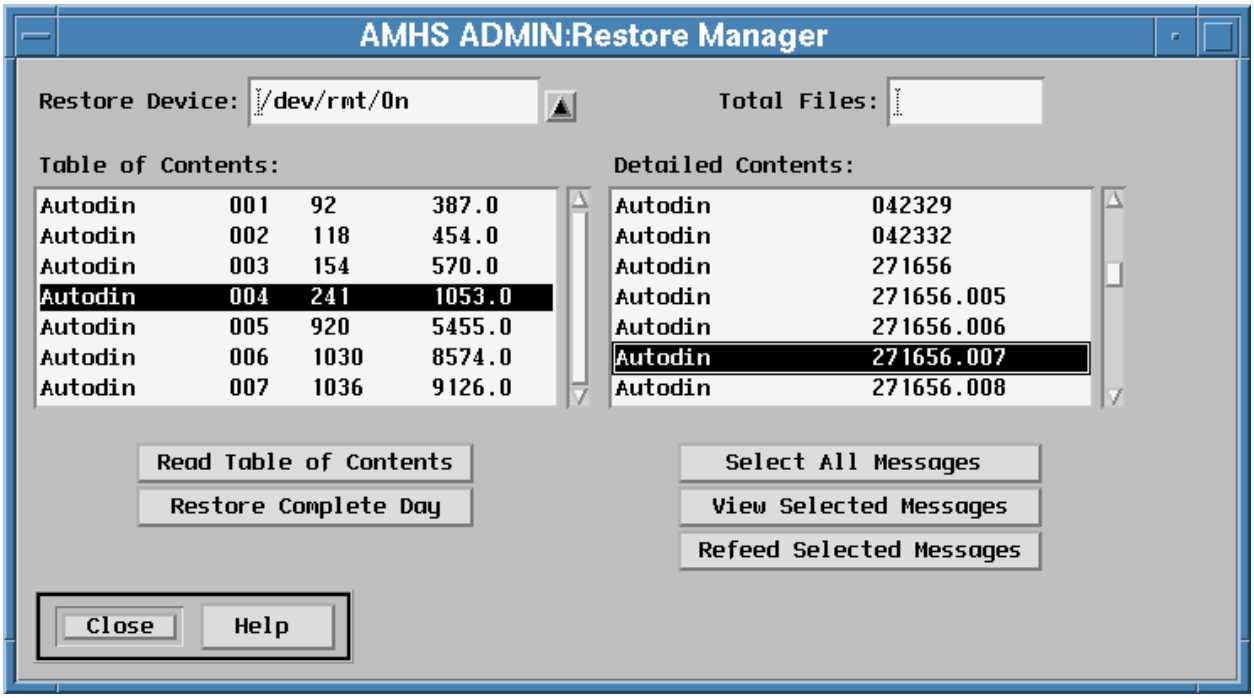


Figure 5-21. Message Restore

5.10.1 Message Restore Process

The message restore allows for restoring single messages or whole days.

TBS

Figure 5-22 Restore Processes

5.11 CUSTOM

This tool is a launch window for special administration scripts. This feature is not implemented in Sys Admin Tools Version 1.3.

5.11.1 Custom Pull-Down

The custom pull-down contains the names of scripts that can be launched from the Sys Admin Tool. The names and functionality of the scripts are personally configured. See Figures 5-23 and 5-24.

TBS

Figure 5-23 Custom Scripts

TBS

Figure 5-24 Sample Script Install

5.12 TOOL CONFIGURATION

There are several items that require configuring before using the AMHS Sys Admin Tool. The following paragraphs provide a detailed explanation of the steps required to configure and operate the tool. Perform these steps only after installing the most current version of the AMHS_SERVER_PATCH. One of the reasons for having a manual setup procedure is to give you familiarity with the files that are required by the Sys Admin Tool. Familiarity with these files will build your confidence when using the Sys Admin Tool.

5.12.1 Configuration File Descriptions

There are several configuration files that the tool manages. These configuration files are located in the **/h/AMHS/Server/data/admin** directory and are visible from any AMHS client workstation (via the network). Each configuration file is responsible for a different tool that is part of the complete Sys Admin Tool (Account Manager, Queue Manager, etc.). Figures 5-25 and 5-26 contain a brief explanation of each of these configuration files:

File Name:	Tool Name:	File Description:
DocumentSources	Queue Manager	Lists all the valid document sources (e.g., AUTODIN)
SecurityValues	Account / DAC Manager	Lists all the valid Topic security groups
DayNumber	Backup Manager	Contains the current Julian day number
SequenceNumber	Backup Manager	Contains a unique number within each Julian day

Figure 5-25. Sys Admin Tools Data Files

File Name:	Tool Name:	File Description:
QueueInformation	Queue Manager	Contains information about each user's queue
UserAccountList	Account Manager	Contains information about each AMHS user account

Figure 5-26. Sys Admin Tools Intermediate Files

These setup procedures concentrate on creating the configuration files required for proper operation of the Sys Admin Tool. The following paragraphs explain each of the previously-mentioned configuration files in detail.

5.12.1.1 DocumentSources

The **DocumentSources** configuration file does not require any setup. In other words, do not edit the file! The file contains a list of the different types of document sources that are available within the AMHS. Currently, as of version 2.0 of the AMHS Server segment, the AMHS supports two document sources. These document sources are AUTODIN and COMEBAK. Changes to this file will be provided on segment patches as more document sources are supported by the AMHS.

The **DocumentSources** file is read by the Queue Manager Tool to assign specific document sources to user queues. Refer to the Queue Manager Tool (Section 5.5) for an explanation on how to configure a user's queue.

5.12.1.2 SecurityValues

The **SecurityValues** configuration file contains a list of Topic security groups. These security groups are used to protect documents within the Topic database. There is a total of 16 possible security groups, each tied to a DAC entry in the **daclist**. At this point, it is not necessary to go into the details about DAC entries. For initial setup purposes, each site will fall into one of the following categories:

(1) Sites that have added or deleted DAC groups by editing the **daclist** file located in the **/h/AMHS/Server/topic/amhs_db/daclist** file. There were nine original DAC groups delivered with your initial AMHS Server segment. The nine initial DAC groups are as follows:

CWP, *fbis*, *exclusive for*, *limdis*, *nato*, *nocon*, *personal for*, *specat*, and *top secret*

It may be necessary to examine the contents of the current **daclist** file in order to determine if entries were added or deleted from this file. Read the header portion of the **daclist** file for a description of what constitutes a DAC entry. Note that adding a code word is not the same as adding a DAC entry.

(2) Sites that are using the **daclist** with no additional DAC groups added. This includes sites that have simply added or deleted code words to one or more of the as-delivered DAC groups.

If your site falls into category (2), the **SecurityValues** file does not need any additional setup. In other words, you can use the file as delivered with the segment. However, you should read the instructions that follow in order to understand the significance of the **SecurityValues** file. If your site falls in the first category, you will need to edit the **SecurityValues** file before using the Sys Admin tool. Below are steps required to edit the **SecurityValues** file.

The **SecurityValues** file, as initially delivered, looks like Figure 5-27 (comment lines omitted).

NOTE: It is important to know that lines 0 through 9 are preset according to the listing in Figure 5-27, and these lines, respectively, *should never be used for any other security group names* than as specified in the figure. However, any line can be deactivated or activated according to the following instructions. Only lines 10 through 15 should be used for other, site-specific security group names. Even then, after creating other security group names in 10 through 15, it is recommended that special care be exercised not to change such names once messages have been thus encoded.

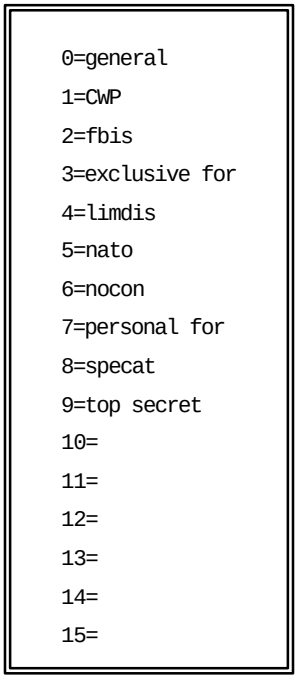


Figure 5-27. DAC Entries List

Each line of the **SecurityValues** file contains a security group name (e.g., **fbis**). Along with each group name is a number that represents the Topic security group for the matching security value. For example, **fbis** has a Topic security group of 2. Any new DAC entries added or deleted from the **daclist** must also be added or deleted from this file. (See Figure 5-28 for an example of adding and deleting.) New DAC entries must be added to the line that matches their Topic security group. In other words, if a new DAC entry called “special” was added to the **daclist**, and this DAC entry has a Topic security group of 11, the following line must be added to your **SecurityValues** file. (See Figure 5-28. For clarity, a majority of the sample file is not shown.)

```
11=special
```

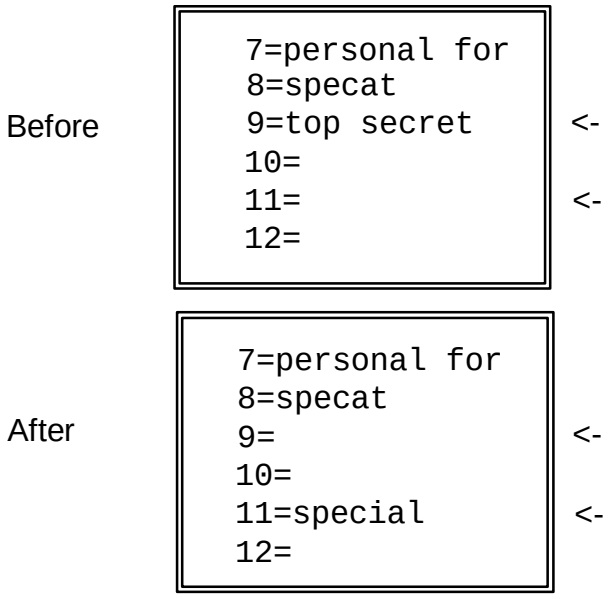


Figure 5-28. Editing DAC List

It may be necessary to familiarize yourself with the **daclist** by reading the comments contained within the actual **daclist** file.

Note: There must be one and only one Topic group number for each DAC entry in the **daclist**. Each message in the Topic database will belong to one and only one of the security groups listed in the **SecurityValues** file.

Likewise, if an entry was deleted from the original **daclist**, that entry must also be deleted from the **SecurityValues** file. For example, the "Top Secret" DAC entry with the Topic security group of 9, shown as "9=top secret" in the upper portion of Figure 5-28, would appear as "9= " in the lower portion of that figure.

The **SecurityValues** file is used by the Account Manager Tool to assign users access to certain messages within the Topic database. This file should not be edited once the changes detailed above have been made. Further changes to this are managed by the DAC Manager Tool. The SecurityValues file is located in: **/h/AMHS/Server/data/admin/SecurityValue** .

5.12.1.3 Day Number

This configuration file contains the current Julian day number. The file is not delivered on the segment installation media, but is created automatically when needed for the first time. Do not create this file; let the system create the file for you. This file is used by the Backup Manager Tool when a Topic publish is performed. Each published partition will have the current day number appended to the partition name. This file is used to retrieve that day number. For example, SAT192V1 is the published AUTODIN publish from day 192. Refer to the section on Topic Publishes in the System Administration Guide for more details.

5.12.1.4 Sequence Number

This configuration file contains a unique sequence number. The file is not delivered on the segment installation media but is created automatically when needed for the first time. Do not create this file; let the system. This file is used by the Backup Manager Tool when a Topic publish is performed. Each published partition will have a unique day sequence number appended to the partition name. This file is used to retrieve that number. For example, SAT192V1 is the published AUTODIN publish from day 192 with a sequence number of 1. Refer to the section on Topic Publishes in the System Administration Guide for more details.

5.12.1.5 User Account List

The **UserAccountList** file contains a list of all the AMHS user accounts. This file is an intermediate file used to manage the Topic password file. The Topic password file is located in the file:

/h/AMHS/Server/topic/amhs_db/password/password

A script is provided to help set up a new **UserAccountList** file. Perform the following steps, those in boldface, to set up the **UserAccountList** file. All commands must be performed as the **amhs_dba** on the AMHS Server workstation. In other words you must be logged on the host **amhserver** using the **amhs_dba** account. If you encounter errors performing these procedures, check the file permissions contents of the AMHS_SERVER_PATCH segment's Scripts and templates directory. The directory and files need an owner of **amhs_dba** and the group of **gccs**.

(1) Make a backup copy of the current Topic password file.

cd /h/AMHS/Server/topic/amhs_db/password

cp password password.backup

(2) Create a new **UserAccountList** file.

cd /h/AMHS_SERVER_PATCH/Scripts

./ CreateAccountList

This command creates a new **UserAccountList** file. The new **UserAccountList** file will be placed in the **/tmp** directory awaiting further processing. At this point, the new **UserAccountList** should be examined to make sure all entries are in the proper format and that each field contains the correct information. As illustrated below, a portion of a sample **UserAccountList** file is provided for field clarification (comment lines are omitted for clarity). Also, a table with a description of each field is provided to help understand the significance of each field. Note that it is very important to make sure this file is accurate before continuing with the setup process.

```
AMHSpos1:0:AMHSADM:AMHSPOS1:amhspos1:NO:0,10:
AMHSpos2:0:AMHSADM:AMHSPOS2::NO:0:
AMHSpos3:0:AMHSADM:AMHSPOS3::NO:0:
AMHSpos6:0:AMHSADM:AMHSPOS6::YES:0:
GCCSGCCSUSER:0:GCCS:GCCSUSER::NO:0:
SUN3pos4:0:AMHSADM:AMHSPOS4::NO:7,3,8,4:
SUN3posX:0:AMHSADM:AMHSPOS5::NO:1:
test3:0:GCCS:ISSO::YES:0:
tester1:0:TESTSUN:TESTER1::YES:0,3,7:
willie:1:::willie:YES:0,1,8,3,7,2,4,5,6,9:
```

The first entry of the sample file (Figure 5-29) conveys the following information (each field is separated by a colon):

Field:	Description:	Data:	Comments:
1	Account Name	AMHSpos1	Will contain your AMHS account names
2	Account Type	0	Will be zero for all your initial accounts
3	Project	AMHSADM	Contains the associated GCCS project for the account
4	Position	AMHSPOS1	Contains the associated GCCS position for the account
5	Password	amhspos1	Contains the account password. Your initial accounts will not have a password in the field. There will be nothing between the semicolons.
6	Delete Permission	NO	Indicates whether the account is allowed to delete messages from the database. Your initial accounts will have a NO in this column.
7	Security Groups	0,10	Contains the Topic security groups the account is allowed to access. Your initial accounts will have a zero in this field.

Figure 5-29 Description of User Account List Fields

Add an entry for the **amhs_dba** account. Append the following entry to the bottom of your **UserAccountList** file:

amhs_dba:1:::habyes:YES:0:

You may replace “habyes” with a new Topic password for the **amhs_dba** account (this is optional). Every other field must appear exactly as it appears in the line above.

At this point it is necessary to edit each account entry by modifying the Topic security groups the account is allowed to access. Initially, all your entries will have a default security value setting of zero. You should change this value to reflect the Topic security groups each account is allowed to access. To do this you have three options: the easiest is to use the Sys Admin Tool to complete the changes; the most challenging is to proceed manually as discussed below; the option of last resort, if this configuration process is not clear to you, is to leave each account with the default value of zero, which equates to “general”, then call the help desk for instructions.

The following is an example of how to make the changes manually:

Suppose you want the “test 3” AMHS account to access the general, CWP and limdis DAC entries (see Section 5.11.1.2 of these instructions for a clarification of DAC entries). The initial **UserAccountList** entry for “test3” looks like the following:

test3:0:GCCS:ISSO::YES:0,**1**,**2**,**3**,....**15:**

Notice the initial security values field contains a zero. The following change must be made to add general, CWP, and limdis access for this user.

test3:0:GCCS:ISSO::YES:0,1,4:

Proceed by making the changes for each of your accounts. Add ALL of the possible available security group values to the **amhs_dba** account. Once these changes are made, the file is complete. The file must be transferred into the working data directory for the server segment. Proceed by executing the following:

(3) Copy the new **UserAccountList** file to the **/h/AMHS/Server/data/admin** directory.

cp /tmp/UserAccountList /h/AMHS/Server/data/admin/

(4) Create a new Topic password file. Start with a blank Topic password file.

cd /h/AMHS_SERVER_PATCH/templates

cp Topic_Password /h/AMHS/Server/topic/amhs_db/password/

(5) Add password entries for each Topic user account.

cd /h/AMHS/Server/Scripts/admin/

Execute the following **AddUser** command for each Topic user account that appears in the newly created **UserAccountList** file. Replace the <username> argument with the actual name of your user.

./AddUser <username>

In other words, the **AddUser** command would be executed eleven times for the sample **UserAccountList** that was previously referenced. The command would be executed ten times for the accounts listed and once for the lastly added **amhs_dba** account. The **AddUser** command reads the information from the **UserAccountList** file and creates a new entry in the Topic password file. The Topic password file will be complete once each account has been processed using the **AddUser** command.

(6) Compile the new Topic password file.

cd /h/AMHS/Server/topic/amhs_db/password/

mkpwd password

cp /h/COTS/Topic/current/bin/topic31.pwd .

(7) Patch the amhs_dba topic preference file.

cd /h/AMHS_SERVER_PATCH/data/amhs_dba

cp topic.prf /h/AMHS/Server/topic/amhs_users/amhs_dba/

This completes the configuration of the Topic password and **UserAccountList** files. From this point forward, the Topic password and the **UserAccountList** file should not be edited unless instructed by a technical support representative. All future changes to these files is completely managed by the Account Manager Tool of the Sys Admin Tool.

5.12.1.6 QueueInformation

The **QueueInformation** file (see Figure 5-30) is used by the Queue Manager Tool. This file is used as an intermediate file that manages the Topic profiles files. This file is prepared similar to the **UserAccountList** file. A portion of a **QueueInformation** is provided for clarification. Note that the comment lines of the file have been omitted to save space.

```
GCCSGCCSUSER:TRUE:pf1:GCCSUSER-i:AUTODIN:GCCS-a:AUTODIN:GCCSUSER-c:COMEBAK:
willie:TRUE:pf1:willie_i:Autodin:willie_a:Autodin:willie_c:Comebak:
AMHSpos1:TRUE:pf1:amhspos1_i:Autodin:amhspos1_a:Autodin:amhspos1_c:Comebak:
AMHSpos2:FALSE::::::::
AMHSpos3:FALSE::::::::
AMHSpos6:FALSE::::::::
test3:FALSE::::::::
SUN3pos4:FALSE::::::::
SUN3posX:FALSE::::::::
tester1:FALSE::::::::
```

The following table provides a brief description of each field of the **QueueInformation** file. The first entry of the sample file (user GCCSGCCSUSER) is illustrated.

Field:	Description:	Data:	Comments:
1	Account Name	GCCSGCCSUSER	Will contain your AMHS account names.
2	Active Flag	TRUE	TRUE for active queues and FALSE for inactive queues.
3	Profiler Name	pf1	Contains the profiler name.
4	Info Topic Name	GCCSUSER-I	Topic used to profile the INFO queue.
5	Info Source	AUTODIN	Document source for the INFO queue.
6	Action Topic Name	GCCS-a	Topic used to profile the ACTION queue.
7	Action Source	AUTODIN	Document source for the ACTION queue.
8	Comeback Topic Name	GCCSUSER-c	Topic used to profile the Comeback queue.
9	Comeback Source	COMEBAK	Document source for the Comeback queue.

Figure 5-30 Queue Information File Description

Execute the following steps to create a new **QueueInformation** file. If you encounter errors performing these procedures, check the file permissions of the contents of the AMHS_SERVER_PATCH segment's Scripts and templates directory. The files need an owner of **amhs_dba** and the group owner of **gccs**.

(1) Prepare a backup copy of your current profiles file.

cd /h/AMHS/Server/topic/amhs_db/pf1topic

cp profiles profiles.backup

(2) Create a new QueueInformation file.

```
cd /h/AMHS_SERVER_PATCH/Scripts
./CreateQueueInformation
```

Executing this script will create a **QueueInformation** file in the /**tmp** directory. This file should be inspected to make certain it agrees with the previously mentioned description of the **QueueInformation** file. There should be one entry for each Topic user account. The script only completes fields 1, 2, 3, 5, 7, and 9. Fields 4, 6, and 8 will have an x, y, and z, respectively.

Completion of fields 4, 6, and 8 of the **QueueInformation** file are made using the Queue Manager Tool. Refer to the section in this manual on using the Queue Manager Tool for an explanation of assigning Topic names to user queues. This step is performed after these setup procedures are complete.

(3) Copy the **QueueInformation** file to the working directory.

```
cp /tmp/QueueInformation /h/AMHS/Server/data/admin/
```

(4) Create a new profiles file. Start with the provided template.

```
cd /h/AMHS_SERVER_PATCH/templates
cp Topic_Profiles /h/AMHS/Server/topic/amhs_db/pf1topic/profiles
```

5.12.1.7 AMHS Server Name

The hostname of the AMHS Server workstation needs to be entered into the **MainWindow.ini** configuration file. This entry will let the Sys Admin Tool know what server should be monitored. Execute the following steps to enter this hostname:

Edit the **MainWindow.ini** configuration file.

```
cd /h/AMHS/Server/data/config

vi MainWindow.ini
```

Enter your hostname on the PROCESSOR_ENTRY line. For purposes of this explanation, we will assume that your AMHS Server hostname is amhsv2. Make the following change:

Before Change:	PROCESSOR_ENTRY=AMHS Server,sun3
After Change:	PROCESSOR_ENTRY=AMHS Server,amhsv2

This information is also included in Section 5.2.1 of this manual. Section 5.11.2.3 contains a detailed explanation of all the entries in the MainWindow.ini configuration file.

5.12.2 Configuration INI Files

These files are used to hold variables and other information that make it possible for the Sys Admin Tools application to be configurable. The Bridge files are ways of passing information and tokens to the applications running or being run by the tools.

5.12.2.1 Admin.ini File

```
# admin.ini  (configuration file)
#
# This is the main configuration file for the
# AMHS Administration tool. This file is used to
# define the location of other configuration files
# and directories.
#
# Make sure you have a complete understanding
# of these values before making any changes.
#

MAIN_WINDOW_FILE=/h/AMHS/Server/data/config/MainWindow.ini
DAC_MANAGER_FILE=/h/AMHS/Server/data/config/DacManager.ini
QUEUE_MANAGER_FILE=/h/AMHS/Server/data/config/QueueManager.ini
CUSTOM_LAUNCH_FILE=/h/AMHS/Server/data/config/CustomLaunch.ini
SCRIPT_LIBRARY_FILE=/h/AMHS/Server/data/config/ScriptLibrary.ini
ACCOUNT_MANAGER_FILE=/h/AMHS/Server/data/config/AccountManager.ini
BACKUP_MANAGER_FILE=/h/AMHS/Server/data/config/ArchiveManager.ini

TEMP_DIRECTORY=/tmp
VARDEF_FILE=/h/AMHS/Server/topic/amhs_db/vardef
LOG_FILENAME=/h/AMHS/Server/topic/amhs_db/log/system_admin
```

5.12.2.2 Script Library.ini File

```
#
# ScriptLibrary.ini  (configuration file)
#
# This file contains a list of the scripts that
# are used for the AMHS Administration tool.

# This script is run at the end of day. The output of the
# script is displayed on the title bar of the application.
```

```
JDAY_FUNCTION=/h/AMHS/Server/Scripts/admin/CurrentDay

# These scripts are used to determine the last transmit and
# receive time for messages. This scripts are run every update
# period for the main window.

XMIT_FUNCTION=/h/AMHS/Server/Scripts/admin/MessageXmitTime
RECV_FUNCTION=/h/AMHS/Server/Scripts/admin/MessageRecvTime
RECV_XMIT_FUNCTION=/h/AMHS/Server/Scripts/admin/MessageRecvXmitTime

# These scripts are used to manage the Topic user accounts. The
# scripts are run whenever new users are added, modified, or
# deleted.

LOGGED_USER_FUNCTION=/h/AMHS/Server/Scripts/admin/CurrentUsers
ADD_USER_FUNCTION=/h/AMHS/Server/Scripts/admin/AddUser
CHANGE_USER_FUNCTION=/h/AMHS/Server/Scripts/admin/ChangeUser
DELETE_USER_FUNCTION=/h/AMHS/Server/Scripts/admin/DeleteUser

# These scripts are used by the Queue Manager tool. This
# script is called whenever the user wants to see a list
# of search criteria (topics).

GET_TOPICS_FUNCTION=/h/AMHS/Server/Scripts/admin/GetTopics

# Scripts for creating queue entries. These are called from
# the queue manager tool.

ADD_QUEUE_INFO_FUNCTION=/h/AMHS/Server/Scripts/admin/AddQueue
CHANGE_QUEUE_INFO_FUNCTION=/h/AMHS/Server/Scripts/admin/ChangeQueue
DELETE_QUEUE_INFO_FUNCTION=/h/AMHS/Server/Scripts/admin/DeleteQueue

# These scripts are used by the backup and restore routines
# that are part of the Backup Manager Tool.

EXTRACT_RECORD_FUNCTION=/h/AMHS/Server/Scripts/admin/ExtractRecord
BACKUP_SIZES_FUNCTION=/h/AMHS/Server/Scripts/admin/BackupSizes
EXECUTE_BACKUP_FUNCTION=/h/AMHS/Server/Scripts/admin/ExecuteBackup
DELETE_MESSAGE_FUNCTION=/h/AMHS/Server/Scripts/admin/DeleteMessage
```

5.12.2.3 Main Window.ini File

```
#
# MainWindow.ini  (configuration file)
#
# This is the main window configuration file for the
# AMHS administration tool.
#
```

```
# The UPDATE_PERIOD is used to configure the number of
# seconds between main window realtime updates.

UPDATE_PERIOD=60

# These entries contain the text used to label the
# main window recv/xmit message information.

XMIT_LABEL=Last Message Transmitted:
RECV_LABEL=Last Message Received:

# This section contains the processor monitoring portion
# configuration information. Field description:
#
#   PROCESSOR_LABEL      Specifies the label above the processor status.
#   PROCESSOR_COUNT      Specified the number of processors to monitor.
#   PROCESSOR_ENTRY      Specifies alias and host to montitor. Specify
#                one for each host to monitor.

PROCESSOR_LABEL=PROCESSOR STATUS:
PROCESSOR_COUNT=1

PROCESSOR_ENTRY=AMHS Server,sun3

# This section contains the configuration information to setup
# the monitoring of message queues. Here is a description:
#
#   QUEUE_LABEL     Specifies the label above the queue status.
#   QUEUE_COUNT     Specified the number of queue entries.
#   QUEUE_ENTRY     Specify the name, queue, and threshold
#                for each queue. Specifiy one entry for each
#                queue to monitor.

QUEUE_LABEL=QUEUE STATUS:
QUEUE_COUNT=7

QUEUE_ENTRY=Emergency Backside Queue,/h/AMHS/Server/sat/autodin/bsq1,2
QUEUE_ENTRY=Flash Backside Queue,/h/AMHS/Server/sat/autodin/bsq2,2
QUEUE_ENTRY=Immediate Backside Queue,/h/AMHS/Server/sat/autodin/bsq3,5
QUEUE_ENTRY=Priority Backside Queue,/h/AMHS/Server/sat/autodin/bsq4,5
QUEUE_ENTRY=Routine Backside Queue,/h/AMHS/Server/sat/autodin/bsq5,10
QUEUE_ENTRY=Transmit Message Queue,/h/AMHS/Server/sat/autodin/xmit,2
QUEUE_ENTRY=Reject Message Queue,/h/AMHS/Server/sat/autodin/reject,2
```

```
# This section contains the configuration information for
# disk monitoring feature. Here is a description:
#
#   DISK_LABEL          Specifies the label above the disk status.
#   DISK_COUNT          Specifies the number of entries to monitor.
#   DISK_ENTRY          Specofy the name, directory, and threshold in
#               percentage. Specify one entry for each directory
#               to monitor.

DISK_LABEL=DISK STATUS:
DISK_COUNT=1

DISK_ENTRY=AMHS File System (% used),/amhs,80
```

5.12.2.4 DAC Manager.ini File

```
#
# DacManager.ini (configuration file)
#
# This is the configuration file for the dac manager
# administration tool.
#
#

DAC_DIR=/h/AMHS/Server/dac
DAC_FILENAME=/h/AMHS/Server/topic/amhs_db/daclist

#
# The WARNING_MESSAGE token is used to control the
# posting of a warning message dialog after every
# dac entry change. Te warning dialog will inform
# users that they must restart the sat_feed and
# cbc_feed programs to have their changes recognized.
#

WARNING_MESSAGE=true
```

5.12.2.5 PLA Manager.ini File

PLA Manager is not implemented in Sys Admin Tools Version 1.3.

5.12.2.6 Queue Manager.ini File

```
#
# QueueManager.ini (configuration file)
#
# This is the configuration file for the queue manager
# administration tool.
#
#
PROFILER_COUNT=1
PROFILER_ENTRY=pf1:/h/AMHS/Server/topic/amhs_db/pf1topic/profiles

DOCUMENT_SOURCES=/h/AMHS/Server/data/admin/DocumentSources
QUEUE_INFORMATION=/h/AMHS/Server/data/admin/QueueInformation

MKUSRTOP=/h/COTS/Topic/current/bin/mkusrtop

USRTOPDIR=/h/AMHS/Server/topic/amhs_db/pf1topic
SYSTOPDIR=/h/AMHS/Server/topic/amhs_db/systopic

DEFAULT_PROFILER=pf1
PRIMARY_SOURCE=AUTODIN
SECONDARY_SOURCE=COMEBAK
```

5.12.2.7 Account Manager.ini File

```
#
# AccountManager.ini (configuration file)
#
# This is the configuration file for the account
# management and DAC portion of the AMHS administration
# toolset.
#
# This contains the name of the UNIX group that
# is used to control release authority.

AMHS_RELEASER=amhs_rel

# This value contains the name of the file used
# to manage the security values.

SECURITY_VALUES=/h/AMHS/Server/data/admin/SecurityValues

# This variable contains the location of the master
# user configuration file.

USER_LIST=/h/AMHS/Server/data/admin/UserAccountList
```

5.12.2.8 Process Manager.ini File

The Process Manager is not implemented in System Admin Tools Version 1.3.

5.12.2.9 Backup and Restore ini File

```
#
# ArchiveManager.ini  (configuration file)
#
# This file contains all the configuration parameters
# for backup and restore tools.
#

# These paramaters are the paths of the support files for
# Archive Manager.

ARCHIVE_INFO=/h/AMHS/Server/data/admin/ArchiveInformation
ARCHIVE_LEVELS=/h/AMHS/Server/data/admin/ArchiveLevels
ARCHIVE_MESSAGES=/h/AMHS/Server/data/admin/BackupSources
DEVICE_LIST=/h/AMHS/Server/data/admin/DeviceList

# This parameter specifies the maximum number of entries
# that will appear in the ArchiveInformation log.

ARCHIVE_LOG_SIZE=20
```

5.12.2.10 Custom (Script and Exec Launcher)

The Custom tool is not implemented in Sys Admin Tools Version 1.3.

5.12.3 Bridge Files

Bridge files are those intermediate files that contain information and/or hold data that is necessary for the tools to interface with other components of the AMHS and Topic processes.

5.12.3.1 Daynumber (Contains The Current Julian Day)

```
009
```

5.12.3.2 Document Sources Data File

```
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
#
# DocumentSources (data file)
#
#    This file contains a list of all the available
# document sources. This file needs to be editted
# whenever new sources are created.
#
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

DSOURCE=Autodin
DSOURCE=Comebak
```

5.12.3.3 Queue Information Intermediate File

```
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
#
# QueueInformation (intermediate file)
#
# This file is used as an intermediate file for
# the Topic profiles file.
#
# The format of each line is as follows:
#   user:enabled:pf:tinfo:sinfo:taction:saction:tcb:scb
# Definitions:
#   user        : topic user account name
#   enabled     : true if queues are enabled
#   pf          : name of the topic profiler
#   tinfo       : name of topic for info messages
#   sinfo       : document source for info messages
#   taction     : name of topic for action messages
#   saction     : document source for action messages
#   tcb : name of topic for comeback messages
#   scb : document source for comeback messages
# NOTE: This file is maintained by the AMHS administration tool

GCCSGCCSUSER:TRUE:pf1:GCCS__GCCSUSER-i:AUTODIN:GCCS__GCCSUSER-
a:AUTODIN:GCCS__GCCSUSER-c:COMEBAK:
willie:TRUE:pf1:willie_i:Autodin:willie_a:Autodin:willie_c:Comebak:
AMHSpos1:TRUE:pf1:amhspos1_i:Autodin:amhspos1_a:Autodin:amhspos1_c:Comebak:
AMHSpos2:TRUE:pf1:amhspos2_i:Autodin:amhspos2_a:Autodin:amhspos2_c:Comebak:
AMHSpos3:FALSE::::::::
AMHSpos6:FALSE::::::::
test3:FALSE::::::::
SUN3pos4:FALSE::::::::
SUN3posX:FALSE::::::::
tester1:FALSE::::::::
```

5.12.3.4 rhosts File

```
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

### rhosts file
sun3 steve
sun2 steve
sun3 willie
sun2 willie
```

5.12.3.5 Security Values Data File

```
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

#
# SecurityValues (data file)
#
#    Master list of security values. This file is
# maitained by the DAC manager tool.
#
#    Each entry has two fields. The first field is the
# security group number and the second field is the security
# group name. There is a 20 character limit on the group names. This
# version of the AMHS Administration Tool is limited to 16 groups
# (as defined by Topic).
#
# NOTE: An exclamation point character '!' is places at the end of
# the name when a name is deleted from the system. All names with
# this character are not available to assign to users.
#

_ _ _ _  _ _ _ _ ___ _ _ _ _ _ _ _ _ _ _ _ _

### Security values - the 1 fix and 15 configurable
0=general
1=CWP
2=fbis
3=exclusive for
4=limdis
5=nato
6=nocon
7=Personal For
8=specat
9=top secret
10=AMHS Test
11=
12=
13=
14=
15=
```

5.12.3.6 User Account List Intermediate File

```
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

#
# UserAccountList (intermediate file)
#
# This file is used as an intermiediate file to
# the Topic password file. This file contains
# information on each Topic account. #
# The format of each line is as follows:
#    name:type:project:position:password:delete:groups
# Definitions:
#    Name     : Topic user account name
#    Type     : Bit specifies type of account (1=user, 0=position)
#    Project  : Assigned GCCS project
#    Position : Assigned GCCS position
#    Password : Topic password
#    Delete   : (YES | NO) to grant delete permission
#    Groups   : List of security values separated by commas
# NOTE: This file is maintained by the AMHS administration
AMHSpos1:0:AMHSADM:AMHSPOS1:amhspos1:NO:0,10:
AMHSpos2:0:AMHSADM:AMHSPOS2::NO:0:
AMHSpos3:0:AMHSADM:AMHSPOS3::NO:0:
AMHSpos6:0:AMHSADM:AMHSPOS6::YES:0:
GCCSGCCSUSER:0:GCCS:GCCSUSER::NO:0:
SUN3pos4:0:AMHSADM:AMHSPOS4::NO:7,3,8,4:
SUN3posX:0:AMHSADM:AMHSPOS5::NO:1:
test3:0:GCCS:ISSO::YES:0:
tester1:0:TESTSUN:TESTER1::YES:0,3,7:
willie:1:::willie:YES:0,1,8,3,7,2,4,5,6,9,10:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
```

5.12.4 Processor Status Definition File

```
# active_spt
#
# This is the active configuration file for for the System
# Process Table. The system process table contains a list of process
# names and host ids which represent the set of processes that the
# Monitor and Control System actively monitors. This file is originally
# based on the process_table file. The process_table will act as the
# master table and this file will be the active table.
#
# The format of the file is as follows:
#
# Any blank line or line containing a "#" in the first character
# position is ignored. Otherwise each line is considered a System
# Process Table entry. Each line has three fields, seperated by "#"
# characters:
#
# server_name#host_name#alias_name#command
#
# The first field is the unique server name used to map entries
# in this table to entries in the control command table. The
# second field is the host name. The alias_name is the name, unique
# to the host, which will be used in status displays, and command
# is the UNIX command string from the UNIX process table which can
# be used to uniquely identify the given proces on a machine.
#
# The Alias_Name field may be up to 25 characters long.
#
# This file is read at System_Executive initialization and is used
# to initialize the System Process Table (SPT)
#
#           Dedicated Processor (edp)
#
u6sysexc#sun2#System Executive#/h/EM/progs/uccs_system_executive
#-------------------------------------------------------------------------
# AMHS Server Segment: List of Processes for the Server to Monitor
#-------------------------------------------------------------------------
satfeed1#sun3#SAT Feed#sat_feed
cbcfeed1#sun3#CBC Feed#cbc_feed
rtmsgqpr#sun3#Queue Profiler#rt prof -PROCNAME pf0
rtmsgpro#sun3#Message Profiler#rt prof -PROCNAME pf1
rtdbserv#sun3#Database Server#rt server -PROCNAME server
cbcmerge#sun3#CBC Message Merger#rt merge -PROCNAME mg4
autodinm#sun3#AUTODIN Message Merger#rt merge -PROCNAME mg1
cbcdatap#sun3#CBC Data Preparation#rt build -PROCNAME dp4
autodind#sun3#AUTODIN Data Preparation#rt build -PROCNAME dp1
#-------------------------------------------------------------------------
```

5.12.5 Login Screen Configuration

The GCCS login screen background (see Figure 5-31) is a GIF file that is always loaded at startup. The User Name and Password window is an overlay and not part of the background layer.

Figure 5-31. Login Screen GIF

If a team is set up with machines dedicated to special operations, or any task that only those machines are to be used for, it is easy to identify them if the Login Screen is changed to display this special use. The Login Screen file is:

/h/EM/libs/xdm/gccslogin.gif

with **gccslogin.gif** as the file that contains the background screen. This file may be renamed and another renamed **gccslogin.gif** for each workstation that needs the special login screen.

5.13 SYSTEM ADMINISTRATOR TASKS

The System Administrator monitors system activities, maintains system files, directs backups, performs restores, and oversees the operation of the primary and secondary AMHS threads.

5.13.1 Restore

This procedure is used to restore files and/or directories from a tape created with **ufsdump**. The restore is performed by the System Administrator or under their supervision. File restore procedures are performed through the use of two unit tape utilities **ufsrestore** and **mt**. The **ufsrestore** utility is used for actually extracting files from tape, while the **mt** is used to position the tape to desired tape image. The **ufsrestore** utility has a convenient interactive option which is recommended Use the standard Sun Solaris procedures for restoring the system to the latest Level 0 backup plus the appropriate Level (1-9) increments.

> **NOTE: There must be an SOP for handling the message traffic that occurred between the last backup and when the system failed, for all known redundant installations.**

A number of possibilities have been presented in this document in the areas where the tools or procedures are discussed. The worst scenario is to just do the UNIX restore and forget about the missing traffic. The second worst is to have the AUTODIN switch refeed the day's traffic, which results in lots of reprocessing for everyone. The best would be to recover the databases and autodin directories before restoring from tape, if possible. The restore process requires some thought and planning and coordinated approval of the ISSO and local approval authority.

The procedure for restoring a file backed-up via the daily backup may be found in the Solaris manual " Administering File Systems" under "Backups and restores of files filesystems." See, also, manpages on **ufsdump** and **ufsrestore** from an Xterm.

5.13.2 AMHS Accounts

AMHS accounts are based on Project/Position Pairs. Project/Position Pairs are created using Profile Manager as part of creating a GCCS User Account as described in Appendix C.1.2 and in the GCCS System Administration Manual GCCS-SAM-2.1. Each Project/Position Pair is associated with one AMHS account. In other words, you must have a Project/Position Pair in order to add a new AMHS user. For an end user to access one of the AMHS accounts, their user ID must be profiled to the Project/Position Pair of the AMHS account. Each AMHS account consists of the following:

(1) Each user must have TOPIC login and a password that is used to gain access to the topic database through the topic retrieval client. The retrieval client is the application used by the end user to read and search messages that are in the database.

(2) Each user must have an AMHS home directory. This is different from their personal UNIX home directory. The AMHS home directories are based on the Project/Position Pair associated with the AMHS account. The home directory contains the personal message queues for the AMHS accounts.

(3) Each AMHS account contains a set of profiles based on the requirements of the Project/Position Pair. Each Project/Position Pair has its own profiling requirements to ensure the end user receives the messages they are interested in reading. Section 5.2 contains a detailed description of AMHS profiles.

The Sys Admin Account Manager Tool performs all of these AMHS tasks for you, and it is important to use the tools for these activities to ensure system/tool synchronization.

5.13.2.1 Adding and Removing UNIX Group Accounts

This is an abstract of the specific instruction cover in Appendix C.1.7. The detailed methods described there are the ones you should use until you are very familiar with the process. This process may be under the control and execution of the site ISSO because it directly affects the GCCS security.

This procedure adds a new UNIX group to the system. One of the ways messages are protected is through UNIX groups. Each message that is fed into the AMHS database is assigned to a specific UNIX group based on the discretionary access controls (DAC) of the message. An end user must be a member of the message's UNIX group and DAC group for them to read the message.

group_name::group_id:users

(1) Add Group Account

(a) With **secman** login, launch Security Manager.

(b) Select: **File -> Group -> New**

A Create Group Window dialog box appears asking for a unique name and number for the group. Use sequential numbers whenever possible.

(2) Remove Group Account

(a) With **secman** login, launch Security Manager.

(b) Select: **File -> Group -> Delete**

A Delete Group Window dialog box appears asking for the name of the group to be deleted.

5.13.3 Deleting Messages from SAT Archive

Upon discovery or notification of unauthorized code-worded messages in the system, these messages must be deleted from the SAT archive. To delete a message from the SAT archive follow these procedures:

(1) Simultaneously press **Alt** and **F1** keys.

(2) Choose the **Search and Retrieval** option.

(3) Select the appropriate archive (receive or transmit).

(4) Choose **Select by Date Time Group** option.

(5) Note the message number (e.g., 161230.287).

(6) Press **<ENTER>** to verify the message is the correct one.

(7) Simultaneously press **Alt** and **F1** keys.

(8) Highlight the **INITIALIZATION FUNCTION** from the screen.

(9) Highlight the **EXIT TO DOS** option and answer **y** .

(10) At J:\AUTODIN type: **cd ARCHIVE\LNNN**
where (L= r [receive] or t [transmit], and NNN = Julian date)

(11) Type: **del {message number}** (e.g., 1961230.287) **.**

(12) Type: **cd \AUTODIN**

(13) Type: **SAT_R**

(14) Highlight the **LOGIN** function from the screen.

(15) Enter **user name** and **password _____**

(16) Highlight the **INITIALIZATION FUNCTIONS**.

(17) Choose the **ON-LINE** function.

5.13.4 Deleting Messages from the Database

Deletion of individual messages is done from the Database Administrator account, **amhs_dba**. Deletion of any message consists of three parts: marking the message as deleted in the topic index, removing the message from the raw messages directory, and removing the message from the DAC directory. These are the steps necessary to remove the message from all three locations.

(1) Marking the message as deleted in the topic index:

(a) Log in as your normal userid to the AMHS Server and follow the procedure below to open the **amhs_dba** account.

xhost +
su - amhs_dba
password (your user password)
runclient
password (topic dba)

(b) Double-click on **Message Browser** from the Query Manager window

(c) Select the message to delete. Note that it may be necessary to perform a retrospective search to locate the message.

(d) Select **File -> Get** info to determine the message filename, message Julian day, and message security group. (Be sure to write this information down because exactly the same information must be used in all three places.)

(e) Select **Edit -> Delete** document. The message is now marked as deleted in the topic index.

(2) Removing the message from the DAC directory area.

(a) Log in to the AMHS Server as the **amhs_dba** account, or open an Xterm window and su to **amhs_dba**.

(b) **cd /h/AMHS/Server/dac/{SSS}/r{JJJ}**
where {SSS} is the security group and {JJJ} is the Julian day.

(c) Type: **rm {filename}**, where {filename} is the message filename. The message is now removed from the DAC directory.

(3) Removing the message from the raw archive directory area.

(a) Login to the AMHS Server as the **amhs_dba** account.

(b) **cd /h/AMHS_SRV/sat/autodin/archive/r{JJJ}**
where {JJJ} is the Julian day.

(c) Type: **rm {filename}**, where {filename} is the message filename. The message is now removed from the raw archive directory.

5.13.5 Refeeding A Message Into The Database

In order to re-enter a message into the AMHS database, perform the following steps.

(1) Identify the message and locate the message filename in the raw archive area.

(2) (Optional) Make a copy the message and correct it. Note that this step is rarely necessary. (See below for more details.)

(3) Copy the message into the special feed directory and run the script for re-entering the message into the database.

These steps are now described in detail.

5.13.5.1 Step 1: Locate The Message Filename In The Raw Archive Area

When a message arrives into the AMHS, it will be written into the raw archive area. The raw archive area consists of receive and transmit directories for each Julian day. Within each of these Julian day directories are UNIX file names which represent the messages received/transmitted for that Julian day. These files will incorporate the messages' Date-Time-Group (DTG) as part of the filename. Therefore, to track down the message you must have its DTG.

(1) Log in to the AMHS Server as the **amhs_dba**.

(2) Type: **cd /h/AMHS/Server/sat/autodin/archive**

(3) Type: **ls**

Look for the directory beginning with **r** and the 3-digit Julian day (e.g., r181).

(4) Type: **cd r{JJJ}**
where **JJJ** is the Julian day identified in (3) above (e.g., cd r181) .

(5) Type **ls {DTG}***
where **DTG** is the message's DTG (e.g., ls 301333* (Note the trailing **Z** in the DTG is omitted.)

(6) The resulting list is of possible files corresponding to the message. View each of the files until you have ascertained the filename for the message. Make a note of the message filename as well as the Julian day.

5.13.5.2 Step 2: (Optional) Correcting The Message

In rare circumstances, a message arrives which has a missing or mismarked security label. This causes the DAC filter to assign an incorrect security group to the message. In this case, the message must be deleted from the database and a corrected copy of the original must be re-entered into the database. The following is used to correct a message to allow re-entry into the database.

(1) Log in to the AMHS Server as the **amhs_dba** .

(2) Type: **cd /h/AMHS/Server/sat/autodin/archive/r{JJJ}**
where JJJ is the Julian day obtained in Step 1 (3) above.

(3) To make a copy of the original message:

Type: **cp {filename} {filename}.new**

(4) Type: **vi {filename}.new**

You will be placed in the vi editor. Use the editing tools to correct the message copy .

NOTE: An incompatibility between the vi editor and the SAT message preamble necessitates that an additional step be performed prior to saving the message: Add two blank spaces to the preamble of the message (the very first line you see in the vi editor).

(5) Save the file by typing ZZ.

5.13.5.3 Step 3: Re-enter the message

The AMHS delivery segment tape includes the "create tokens" script for entering pre-canned messages into the database.

(1) Log in to the AMHS Server as the **amhs_dba**

cd /h/AMHS/Server/Scripts/admin
Create Token f181/ (filename)

2) Verify the new message was re-entered with no problems.

cd /h/AMHS/Server/topic/amhs_db/log
tail -30f sat_{JJJ}.log
where JJJ is today's Julian day. A scrolling display of the **sat_feed** log file will appear on the window. Verify the filename you entered was processed with no problems.

3) Type: **<Control>-C** to return to the command prompt.

5.13.6 Accessing a Message in Reject Queue

There are two methods used to access messages in the Reject queue, perhaps to determine the reason for rejection.

(1) From the SAT Terminal:

(a) Press **Alt F1** while the SAT processor main window is displayed. The Message Preparation & Retrieval menu window is displayed.

(b) Select **Message Retrieval and Edit** (default). This will display the Directories list.

(c) Select the **Reject Directory**. All rejected messages will be displayed.

(e) Select the message you want to display**.** The selected message will be displayed in its entirety.

(f) Press **F10**. Select the option you want, e.g., transmit or save the message to the system files. Press **<ESC>** to cancel all previous actions.

(2) From an Xterm window on a GCCS workstation:

(a) Launch an Xterm window.

(b) Type: **rlogin amhserver -l amhs_dba**

(c) **Password**_____

(d) Type: **cd /h/AMHS_SRV/sat/autodin/reject**

This will put you in the reject queue.

(e) Type: **ls**

This will display the rejected messages.

(f) Select the rejected message in question.

(g) Type: **cat** then cut and paste by using the middle mouse button.

(h) Look for a reject error message at the top of the screen.

(i) Notify user of any rejected messages.

5.13.7 Updating and Activating Queues

These steps are in concert with Query Manager.

5.13.7.1 Updating Queues

These are the procedural steps to edit/update queues:

(1) Remotely log in to the AMHS Server as the **amhs_dba** by typing:

```
xhost +
rlogin  amhserver  -l  amhs_dba
```

(2) Activate the TOPIC client as the Database Administrator by typing:

```
runclient
```

(3) Use the TOPIC editing features from within the TOPIC client to make additions/edits as necessary. Save the edits by selecting **File -> Save Topics** .

Exit the TOPIC client by selecting **File -> Exit** .

The convention used at many sites for naming topics which are profile queries is as follows:

{Project}__{Position}-a for Action
{Project}__{Position}-i for Info
{Project}__{Position}-c for Comeback Copy,
e.g., (ETCC__ECJ3XO-a).

Children and other types of subtrees, if hierarchical reusable modules are part of the design strategy, should be given names such that their functionality is easy to recognize when seen from the query list window in Topic Editor.

Additional information on how to build topics from within the TOPIC Editor can be found in the Theory of Operation portion of this document, Section 2.4.2, and in the TOPIC Database Administrator's Guide, Section 8.

Now assign topics to user queues using the Queue Manager tool.

(4) If you want to write the profiler topics out in outline form (ASCII) to a save file for future reference type:

cd /h/AMHS/Server/topic/amhs_db/

mkusrtop -s systopic -u pf1topic -fullotl prof.ot1.{date}

where date is today's date in YYMMDDHHMM format

It is important to keep the Binary and ASCII versions of the Topic Query database in synchronization because you will need them if it is ever necessary to reconstruct one or the other.

5.13.7.2 Activating Queues

Use the AMHS Administration Queue Manager Tool to assign and activate user queues. Note that it is assumed the underlying action queue criteria have already been established (see the preceding section entitled "Updating Action Queues").

5.13.8 Saving Topic Query Information

There is one profiles file for each of the profiler processes. Each profiler's profiles file is located in the subdirectory for the respective profiler. For example, the profiles file for the **pf1** profiler is located in the **/h/AMHS/Server/topic/amhs_db/pf1topic** subdirectory.

There is only one Topic Query Database for all the profilers. The Topic Query Database is located in the **/h/AMHS/Server/topic/amhs_db/pf1topic/usrtop** directory. The Topic Query Database is a binary set of files located in various subdirectories within the **usrtop** directory. The Topic Query Database is edited by the **amhs_dba** using the Topic Query Editor. Saved Topic queries are written to the binary files that make up the Topic Query Database. There is only one Topic Query Database and the size of the database does not affect the performance of the individual profilers. (See Figure 5-32.)

Since the Topic Query Database is in binary form, the AMHS Administrator should make periodic copies of the database in ASCII form. The **mkusrtop** command is used to export the Topic Query Database into ASCII form. Execute the following to backup the Topic Query Database:

```
cd /h/AMHS/Server /topic/amhs_db
mkusrtop -s systopic -u pf1topic -fullotl pf1topic/pf1topic.otl
```

When the time comes that it is necessary to load the Topic Query Database from scratch, the following commands will load the Topic Query Database from a source ASCII file:

```
cd /h/AMHS/Server /topic/amhs_db
mkusrtop -0  -s systopic -u pf1topic -o pf1topic/pf1topic.otl
```

There is no need to shut down any of the profilers to execute this command. The command used to import the Topic Query Database from ASCII to Binary form is:

```
cd /h/AMHS/Server/topic/amhs_d
mkusrtop -0 -s systopic -u pf1topic -0 pf1topic/pf1topic.otl
```

The previous commands produce full copies of the Topic Query Database. In other words, the export function will export the entire Topic Query Database to an ASCII file. Likewise, the import command will destroy the Topic Query Database and reload the database from the specified ASCII file. Remember, the profiler process (**pf1**, **pf2**, ...) is the process that reads the Topic Query Database. If there are problems with your Topic Query Database, errors will begin to appear in the profiler's audit log. This may be an indication that it is time to reload your Topic Query Database. There is no telling when and if you will need the ASCII files, but periodically creating copies using the export command will prepare you for an emergency import.

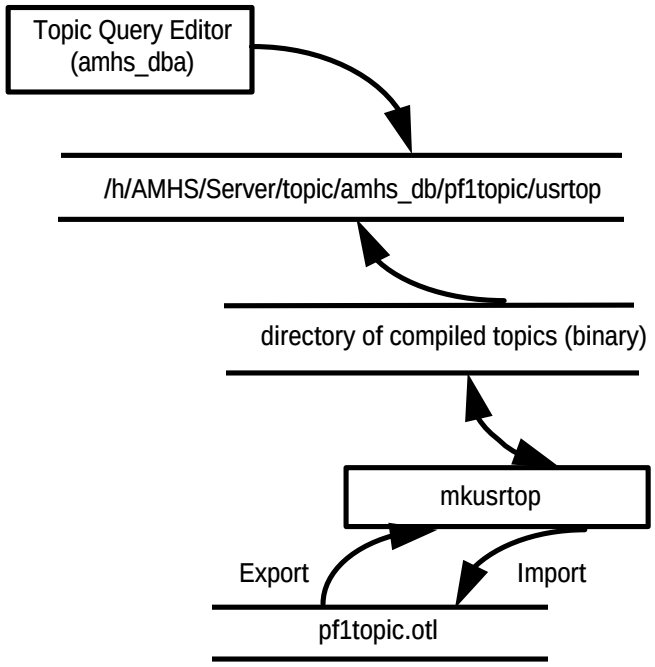


Figure 5-32. Topic OTL File Processing

5.13.9 Configuring TOPIC User Profiles

The AMHS profiler process(es) is denoted by an **rt prof**' process in the UNIX process table. There may be more than one profiler process running at any one time. Each instance of a profiler process is distinguished by the profiler name. For example, the first profiler is named **pf1**. Therefore, the first profiler's complete UNIX process name is **rt prof pf1**. Subsequent profilers are named **pf2**, **pf3**, etc. Additional profile processes are created by the System Administrator

in order to minimize the amount of processing required by any one individual profiler. In other words, when a profiler process becomes too slow from having a large profile database, a new profiler process is created, and the profile database is divided amongst the two profiler processes. Refer to Section 2.4 in this document or chapters 8 and 9 in the TOPIC Real-Time Administrator Guide for the details on adding new profilers.

TOPIC profiles files are used by the TOPIC Real-Time profiler processes to automatically execute queries against incoming documents. The queries consist of topic Boolean plus expressions containing topics, words and phrases. The query statements contained in the profiles are designed to perform various tasks when the expression is satisfied. Each Boolean expression is individually constructed for each AMHS account.

In general, when an expression is satisfied, the profiler performs the following tasks for the AMHS user:

(1) Forwards the message into one or more of the user's message queues.

(2) Updates the message delivery record to contain the recipient's account name.

(3) Notifies the users when the task is complete.

The TOPIC profiles files are located in subdirectories within the **/h/AMHS/Server/topic /amhs_db** directory on the AMHS Server. Each profiler has its own directory within this directory that contains a profiles file for the profiler process. For example, for **pf1** profilers, profiles file are stored in the **/h/AMHS/Server/topic/amhs_db/pf1topic** subdirectory. The **pf2** profiler's profiles and outline file are stored in the **/h/AMHS/Server/topic/amhs_db/pf2topic** subdirectory. Refer to the TOPIC Real-Time Administrator's Guide and the actual operational profiles and outline files for information about the contents of the profiles and outline files.

The profiler processes must be shut down before the profiles and outline files are edited. For example, the **pf3** profiler must be shut down before the profiles and outline files are edited. Once the files are edited, the files must be compiled using TOPIC's **mkusrtop** command. Execute the following command to compile the **pf1** profiler's outline file:

```
cd /h/AMHS/Server /topic/amhs_db
mkusrtop -0  -s systopic -u pf1topic -o pf1topic/pf1topic.otl
```

The **pf1** profiler can be restarted once the **mkusrtop** command completes successfully. If the command fails, the problem is most likely a syntax error in the files. Refer to the TOPIC Real-Time Administrator's Guide for further details on TOPIC central profilers.

5.13.10 Discretionary Access Control

The DAC Manager is a user interactive program designed to provide a convenient method of altering the site-specific **daclist** file. It ensures the proper format of such a file and is therefore vital to maintaining its integrity. (See **daclist** in **rtdb** directory for more information.)

5.13.10.1 Terminology

DAC types have two distinct characteristics associated with them: the type's access information and its keywords. The access information consists of the DAC type's name, the search method used, the DAC directory, UNIX group and protection, and the TOPIC mask associated with the particular DAC. Keywords must also be included.

5.13.10.1.1 DAC Name

Naming the DAC type merely provides a convenient way of referring to it. The name generally reflects the security category; for example, “SPECAT” or "Top Secret".

5.13.10.1.2 Search Method

This is the least intuitive of all the DAC access information. The search method can have one of three possible values: a precedence search, an "unless" search or a combination search.

A precedence search method indicates to the **type_msg** routine that when a keyword belonging to the type is found, the message should immediately be assigned the access associated with that type and the routine should stop. The order in which the type definitions are listed in the **dac_list** is of utmost importance. The type possessing the highest security level must be first among those also using the precedence search to ensure that users cannot access messages with higher security levels than they are authorized.

A filter search method is just a special case of the precedence search. If a message is found to have a keyword associated with a type using the filter search method, the message will be assigned the access associated with that type *unless* the message is found to also contain a keyword associated with another pre-defined DAC type. In the case where the message contains both keywords, the access will be that corresponding to the second filter DAC type. Since the search is recursive, a DAC type defined to be the filter type for one type can also have a filter DAC associated with it. Currently, no validation is done to ensure that these filter definitions are not circular. It is left to the person defining these filters to ensure that this does not happen.

By contrast, a combination search method indicates that when a keyword belonging to the type is found, the TOPIC mask associated with the type should be saved. The search continues, and the masks of all combination search types which incurred "hits" are "or-ed" together. This provides a way for users from two access groups to read the same message, as long as the message contains at least one keyword from each of the types.

It is important to note that the search method is a characteristic of each individual DAC type. It is possible to have a mixture of these two methods within the **dac_list**. Note that whenever a message incurs a hit on a precedence search type, the message belongs to that type regardless of any previous hits on combination search types. Precedence searches always outrank combination search types. Thus, the order of a precedence search type is only important relative to other precedence search types.

5.13.10.1.3 Directory

This describes where in the system the messages assigned to this type should be stored. The directory is relative to the **/h/AMHS/Server/dac** directory structure.

5.13.10.1.4 Group

The UNIX group is the name of the group the messages will be assigned. (All messages are owned by the **amhs_dba**). Error checking is done to ensure that the group entered is defined in the UNIX group file. The **sat_feed** and **cbc_feed** will not start, if any of the DAC groups are not defined.

5.13.10.1.5 Protection

UNIX protection is given to the messages stored in the directory. It should be defined the standard way—three digits. The permissions are 770.

5.13.10.1.6 Mask

The mask is the TOPIC access mask which determines which users have access to the message within the Topic database. If a user has privilege to messages with a particular mask, the messages associated with this mask will be displayable in the results list of the user's queries. However, the user must also have UNIX permission to actually read the message. This mask is associated with the topic group and is an integer between “0" and “15".

5.13.10.1.7 Keywords

The keywords are simply the words or phrases that the **type_msg** routine will search for to classify the message. **type_msg** ignores all blanks and is not case sensitive. No characters that are not explicitly expected to occur in a message should ever be used to define a keyword. The length of a single keyword is unlimited. There are three types of keywords: stem, literal and regular expressions.

(1) Stem keywords: A stem keyword is defined by surrounding the keyword in single quotes: 'stem'. All blanks will be stripped from the word/phrase you enter. The messages will then be searched for occurrences of the sequence of letters (or numbers) you entered separated by any number of blanks. For example, if you define 'word' to be a stem keyword, a message containing any of the following will be considered a hit: "**word**", "s**word**sman", "ne**w ord**ers", "**w o r d**".

(2) Literal keywords: A literal keyword is defined by surrounding the keyword in double quotes: "literal". All blanks will be preserved. The messages will then be searched for occurrences of the exact sequence of letters/numbers you entered. The first character either must be preceded in the message by a space or it must be at the beginning of a line. The last character of a literal must be followed by a space or must be at the end of a line. For example, if you define "word" to be a stem keyword, a message containing the following will be considered a hit: "a **word** is".

(3) Regular Expression keywords: A regular expression keyword is defined by surrounding the keyword in single back quotes: `reg exp`. The messages will then be searched for occurrences of the regular expression you entered. The available regular expressions are those used with the UNIX editor and are attached at the end of this reference.

5.14 AMHS SYSTEM ADMINISTRATION AS amhs_dba

The Database Administrator oversees the operations of the AMHS TOPIC Database. The duties include user management, message management, and management of the individual processes which make up the AMHS Database.

5.14.1 Database Processes

To be fully operational, the database system requires that nine (9) independent processes be up and running. Figure 5-33 lists the processes and describes their functions.

PROCESS NAME	FUNCTION
rt server	Database server.
rt build - dp1	"Document preparation" of inbound AUTODIN messages to load them into the message database.
rt merge - mg1	Merge process to consolidate inbound AUTODIN messages in the message database.
rt build - dp4	"Document preparation" of the "comeback" copies of messages to load them into the message database.
rt merge - mg4	Merge process to consolidate the "comeback" copies of messages in the message database.
rt prof pf0	The API process that determines the distribution of messages.
rt prof pf1	The process which appends a delivery record to the incoming AUTODIN message.
sat_feed	Initial processing of inbound messages received by the Standard Automated Terminal (SAT).
cbc_feed	Initial processing of the "comeback" copies of outbound messages.

Figure 5-33. Database Processes

5.14.2 Topic Real-Time Process Management

The UNIX SCRIPT **topic_cmd** is a set of subcommands that ease the operation and management of the TOPIC system. The functionality of each of these commands is fairly self-evident.

```
AMHS SYSTEM ADMIN AND DATABASE ADMIN COMMANDS MAIN MENU

 STARTUP COMMANDS:
         1) All AMHS Processes          11) SAT Feed    21) Merge 1 (mg1)
         2) Topic Server Process        12) CBC Feed    22) Merge 4 (mg4)
         3) Topic Profilers             13)             23)
         4) Sat Feed & Processes        14)             24)
         5) CBC Feed & Processes        15)             25) Build 1 (dp1)
         6)                             16)             26) Build 4 (dp4)
         7)                             17)             27)
         8)                             18)             28)

 SHUTDOWN COMMANDS:
         31) All AMHS Processes         41) SAT Feed    51) Merge 1 (mg1)
         32) Topic Server Process       42) CBC Feed    52) Merge 4 (mg4)
         33) Topic Profilers            43)             53)
         34) Sat Feed & Processes       44)             54)
         35) CBC Feed & Processes       45)             55) Build 1 (dp1)
         36)                            46)             56) Build 4 (dp4)
         37)                            47)             57)
         38)                            48)             58)

 TOPIC UPDATE & EDIT COMMANDS:
         61) Edit Profiles File         66) Update Profiles File
         62) Edit Password File         67) Update Password File
         63) Edit Systopic Files        68) Update Systopic File
         64) Edit Prftopic File         69) Update Prftopic File
         65) Add New Topic User         70) Update User Topics File

 MONITOR COMMANDS:
         71) Messages Received at the SAT
         72) Backside Message Queues
         73) Outgoing Message Queue
         74) Reject Message Queue
         75) Time Elapsed Since Last Message Received
         76)

 OTHER COMMANDS:
         81) Create new Profiler
         82)
         83)

         0) Topic Status

Enter your option [. to exit] 6:45 AM
```

Several of these commands are included in the Sys Admin Tools. When there is a conflict, the tools take the lead and disable commands 61-70 on this menu.

5.14.3 Publishing Real-Time Partitions

The “publish” script is used for organizing collections of AMHS messages into Topic partitions. It is scheduled to run once a day as an entry in the crontab file. When executed, the publish script will consolidate all of the messages received during that day into a single static partition:

45 23 * /h/AMHS/Server/Scripts/admin/DailyPublish >/dcu/console**

This process is described in detail in Section 2.3.13.

5.15 AUTODIN TERMINAL (SAT/CBT) ADMINISTRATION

Refer to Cavalier SAT and CBT Manuals for more detailed instructions.

The SAT provides receipt and transmission of messages via the AUTODIN. This interface provides secure, reliable access to worldwide command and control centers as well as a variety of other defense and civilian agencies. The SAT is connected to AUTODIN via the TCC.
The SAT resides on a PC-compatible 80386 (or higher)-class machine. The SAT PC has a Network Interface Card (NIC) for Ethernet LAN access, and a GMM Communications Control Processor (CCP) serial communications card for access to AUTODIN. The software suite the SAT must have installed includes the device drivers required for LAN access and the SAT software that performs all the necessary processing to meet the requirements for connectivity to AUTODIN.

5.15.1 SAT Startup

During normal boot-up processing, the CPU boots from the hard disk in the SAT and runs the DOS-standard CONFIG.SYS and AUTOEXEC.BAT files. Then PCNFS Mounts to the AMHS Server.

Once the SAT display contains the Communications Status Display, Communications Control Menu and Hot Keys Display, the SAT is loaded onto the PC and may be initialized. If operator logon is enabled for the SAT, the operator must log on with a valid user name and password prior to initializing the system. Once logged on, initialize the SAT by using the arrow keys and a carriage return as follows:

(1) Select **Initialization Functions** from the Communications Control Menu.

(2) Select **On-Line** from the Initialization Menu.

The SAT will indicate that it is "On-Line" in the upper right-hand corner of the Communications Status Display. For further details concerning SAT initialization, consult the SAT User's Manual.

Should the operator wish to restart the SAT without rebooting, enter the following sequence at the DOS prompt:

```
J:\
CD \AUTODIN
SAT_R
```

This reloads the SAT software into the PC. Following the procedure detailed above, place the SAT on-line.

5.15.2 SAT Shutdown

The SAT should be shut down only as required for maintenance functions, such as site-specific PLA and Address Indicator Group (AIG) table updates or as directed by the appropriate authority. Under these circumstances it is preferable to wait for the SAT to be inactive, i.e., not receiving or sending any traffic. This is indicated by lack of activity on the transmit/receive lines display area in the upper right corner of the SAT Communications Status Display.

(1) Select **Communications Functions** from the Communications Control Menu.

(2) Select **Receive Stop on ETX** .

(3) Select **Communications Functions** from the Communications Control Menu.

(4) Select **Transmit Stop on ETX** .

This forces the SAT and its corresponding circuit to stop transmission and reception following completion of the receipt and/or transmission of the current message(s). Once there no longer is activity on the transmit/receive lines display, exit the SAT. To exit the SAT:

(5) Select **Initialization Functions** from the Communications Control Menu.

(6) Select **Exit To DOS**.

The link has been disabled and the normal DOS prompt is displayed. For further details concerning SAT shutdown, consult the SAT User's Manual.

5.15.3 SAT Monitoring

During normal system operation, the SAT should operate stand-alone, that is, operator intervention will not be required for receipt and transmission of traffic. Whenever operator intervention is required, the SAT emits an audible alarm and a highlighted field will indicate the cause of the alarm to the operator. The cause for intervention may either be normal operation, as in the notification of high precedence traffic, or failure notification, as in disk error, loss of frame or three back-to-back Negative Acknowledgments (NAKs) from the receiving system [Automated Multimedia Exchange (AMME), Air Force Automated Message Processing Exchange (AFAMPE), Message Distribution Terminal (MDT)] or other AUTODIN Switching Center equipment]. To clear an alarm, consult the site-specific SOP if the alarm is due to a failure notification.

During initial system operation, all received SAT logs and audit trails are printed to an attached laser printer. These printouts should be maintained in a SAT log book for a duration as specified by the site-specific SOP. During system operation, the SAT requires an active monitor only when an alarm is sounding. The software is self-monitoring, as required by/for interface terminals, to assure message integrity across the communications link and within the SAT.

Consult the site-specific SOP for failure conditions requiring operation intervention and appropriate responses. At the close of every processing day (Julian day), the SAT prints out a daily log for the traffic processed during that day if a printer is attached. These printouts should be maintained in the SAT log book for the duration specified by the site SOPs.

5.15.4 SAT Message Operations

The SAT supports a variety of ways to generate or source a message destined for the local AUTODIN switch. Messages may be generated from scratch or they may be read-in either from a floppy or the form directory. The methods discussed include submitting messages from a floppy disk, generating/editing and transmitting a message, generating a DD173 message, printing and canceling a message.

5.15.4.1 Submitting Messages From Floppy Disk

Should the operator wish to submit a message from a floppy, the following procedure should be observed:

(1) Insert the floppy containing message(s) into drive A.

(2) Select **TRANSMIT MESSAGE** from the Communications Control Menu.

(3) Select **SINGLE MESSAGE** (if sending one message only) or **MULTIPLES MESSAGES** from the Transmit Message Menu.

(4) Select **FLOPPY DRIVE A** from the Directory/Device Selection Menu.

(5) Select message(s) to be submitted from the directory display using the arrow and space keys (space = toggle for select/deselect).

(6) Press **END** key when finished.

5.15.4.2 Editing and Transmitting Messages

Should the operator wish to edit and transmit a message from the SAT, the following procedure should be observed:

(1) Press **<ALT><F1>** .

(2) Select **MESSAGE RETRIEVAL AND EDIT** from the Message Preparation & Retrieval Menu.

(3) Select **FLOPPY DRIVE A** (or appropriate directory) from the Directory/Device Selection menu.

(4) Select **MSG** from the directory display.

(5) Edit the message.

(6) Press **<F10>** key when finished.

(7) Select **T** to transmit the message.

5.15.4.3 Generating DD173 Messages

Should the operator wish to generate a DD173 message from the SAT, the following procedure should be observed:

(1) Press **<ALT><F1>** .

(2) Select **DD173 MESSAGE PREPARATION** from the Message Preparation and Retrieval Menu.

(3) Fill in the required fields as prompted by the software.

(4) Enter **<END>** to proceed to the next page.

(5) Edit the message.

(6) Press **<F10>** key when finished.

(7) Select **F** to file the message.

5.15.4.4 Printing a Message

Should the operator wish to print a message or group of messages from the SAT, the following procedure should be observed:

(1) Press **<ALT><F1>** .

(2) Select **PRINT FILE(s)** from the Message Preparation and Retrieval Menu.

(3) Select **FILE DIRECTORY** (or appropriate directory) from the Directory/Device Selection Menu.

(4) Select **MSG**s from the directory display.

(5) Press **<END>** key when finished.

5.15.4.5 Canceling a Message

Should the operator wish to cancel a message during transmission from the SAT, the following procedure should be observed:

(1) Press **<F2>** .

5.15.5 SAT Stand-alone Operation

The SAT is capable of stand-alone operation. In emergency situations this might be necessary. The documentation supplied with the equipment describes the steps to configure and operate the SAT as a stand-alone AUTODIN message system.

5.15.6 SAT Recovery Procedures

Several classes of problems may occur with respect to the SAT system. These are identified by area and discussed in the following subparagraphs.

5.15.6.1 Circuit Communications

Problems in the circuit can occur for a variety of reasons including: loss of crypto synchronization, excessive noise on the circuit and incorrect configuration of the SAT hardware or software. These types of problems will manifest themselves as numerous communications alarms generated by the SAT, with errors such as the **3NK** indicator display.

To ensure that the SAT has been correctly configured, the operator may place the SAT in self-test mode and manually submit message traffic to test the system configuration without actually using the remote switch. For sites that prevent transmission from the SAT for security reasons, self-test mode is disabled. However, to put the SAT in self-test mode for most sites, use the following procedure.

If the SAT is already on-line, the software must be shut down. (Note that the Officer-in-Charge should concur with a decision to shut down the link for any reason). Once shut down, restart the software. Once the Communications Status Display is presented, the operator will:

(1) Select **Initialization Functions** from the Communications Control Menu.

(2) Select **SELF-TEST** from the Initialization Menu using the arrows and carriage return key.

The SAT will indicate "Self-Test" in the upper right-hand corner of the Communications Status Display. The SAT will be operational, though it will not be communicating with the AUTODIN switch.

All messages transmitted will be looped back to the SAT's own receive directory.

For further details concerning SAT initialization, consult the SAT User's Manual. To return the SAT to "On-Line", exit the SAT software and restart.

If the SAT appears to work in self-test mode, the circuit and its components (such as the crypto) should be checked for problems.

5.15.6.2 LAN Communications

Problems with LAN communications will manifest themselves quickly with a long, audible alarm, and a special SAT display indicating **DISK COMMUNICATIONS CUTOFF.** The SAT software will "hang" the PC, that is, the only possible response will be to reboot the system. In general, this will be caused by loss of the connection to the LAN or loss of the Ethernet host (AMHS Server) due to power failure or reboot. The SAT may be reconfigured to work without LAN communications should the problem not be immediately evident and the situation require continued availability of the link.

SAT ERROR	DESCRIPTION	FIX
DISK COMMUNICATIONS CUTOFF	SAT can no longer access the AMHS disk.	1. Verify transceiver cable is plugged into SAT. 2. Verify transceiver cable is plugged into AMHS. 3. Verify AMHS is powered on and through reboot processing (i.e., has a login prompt). 4. At DOS prompt: **Error reading drive J: (Abort, Retry, Ignore),** enter **R**. 5. At J:\ prompt, enter **Startup**.
Cannot write file	SAT unable to write edited file to the disk. Either the disk is full or unavailable.	1. Check disk space on E: by issuing a **DIR** at the J: prompt. 2. If less than 1 MB is available, contact the AMHS administrator to free up disk space. 3. If disk space is greater than 1 MB, verify transceiver cable is plugged into SAT. 4. Verify transceiver cable is plugged into AMHS. 5. Verify AMHS is powered on and through reboot processing (i.e., has a login prompt). 6. At DOS prompt: **Error reading drive J: (Abort, Retry, Ignore),** enter **R**. 7. At J: prompt, enter **Startup**.
Reading file: NNNN	SAT can no longer access the AMHS disk.	1. Verify transceiver cable is plugged into SAT. 2. Verify transceiver cable is plugged into AMHS. 3. Verify AMHS is powered on and through reboot processing (i.e., has a login prompt). 4. At DOS prompt: **Error reading drive J: (Abort, Retry, Ignore),** enter R. 5. At J: prompt, enter **Startup**.
Writing file: NNNN	SAT can no longer access the AMHS disk.	1. Verify transceiver cable is plugged into SAT. 2. Verify transceiver cable is plugged into AMHS. 3. Verify AMHS is powered on and through reboot processing (i.e., has a login prompt). 4. At DOS prompt: **Error reading drive J: (Abort, Retry, Ignore),** enter R. 5. At J: prompt, enter **Startup**.
Cannot access directory	SAT cannot access specified directory.	1. Verify transceiver cable is plugged into SAT. 2. Verify transceiver cable is plugged into AMHS. 3. Verify AMHS is powered on and through reboot processing (i.e., has a login prompt). 4. At DOS prompt: **Error reading drive J: (Abort, Retry, Ignore),** enter **R**. 5. At J: prompt, enter **Startup**.

SAT ERROR	DESCRIPTION	FIX
Creating Directory ESC: Abort ANY KEY: retry	SAT could not create a directory.	1. Check disk space on J: by issuing a **DIR** at the J:\ prompt. 2. If less than 1 MB is available, contact the AMHS administrator to free up disk space. 3. If disk space is greater than 1 MB, contact AMHS administrator to verify that SAT has write access to the J: drive. 4. Reissue command.
Insufficient memory	SAT could not initialize/continue due to insufficient available memory in the PC.	1. Examine the **config.sys** file for the system. 2. Remove additional drivers currently installed on the system and not required for the SAT. 3. Restart the SAT.
Operator aborted input	This error is given when the operator escapes from generating a message.	None.
Must initialize system first	SAT must either be in "SELF-TEST" or "ON-LINE" prior to issuing any other SAT commands.	1. Select **Initialization Functions** from the Communications Control Menu. 2. Select either **ON-LINE** or **SELF-TEST.**
Transmit single message busy	Generated when the operator wishes to transmit another message while the SAT is currently transmitting.	1. Wait a few seconds for the SAT to complete transmission of the current message. 2. Reenter command.
Transmit multiple messages busy	Generated when the operator wishes to transmit another message while the SAT is currently transmitting.	1. Wait a few seconds for the SAT to complete transmission of the current message set. 2. Reenter command.
Missing Plain Language Address file - PLA_LINK.?	Generated when the PLA_LINK.R file gets deleted from the SAT directory, **J:\AUTODIN.**	1. Restore the original PLA_LINK.R file from installation diskette. 2. On J:, **cd \AUTODIN**. 3. Enter **BUILDPLA**.
No more file handles	Generated when too many open files already exist.	1. Edit config.sys to increase file setting as follows: **FILES = 50** 2. Reboot the PC. 3. Restart the SAT software.

5.15.7 SAT PLA/AIG Tables

Problems with the format of an outbound message submitted via the LAN may cause the SAT or the Mode I host to reject an outbound message. In such cases, the operator must take the necessary action to allow the message to be transmitted, or notify the Designated Release Authority that the message was not transmitted and why. Cases and appropriate action responses are outlined below.

SAT ERROR	DESCRIPTION	FIX
PLA not in table	SAT received a TO or INFO line addressee which does not exist in its current PLA table.	1. Verify PLA is valid. 2. Obtain Routing Indicator (RI) for the PLA. 3. During system maintenance, i.e., SAT not operational, update the PLA list used by the SAT.
Unable to open AIG file: AIGNNNN	SAT received a TO or INFO line AIG which does not have a corresponding AIG file in the SAT directory.	1. Verify AIG is valid. 2. Obtain list of RIs for the AIG. 3. During system maintenance, i.e., SAT not operational, create a valid AIG file in the SAT directory.
Invalid DD173 / JANAP 128 / ACP 126 format: NNNN	Message read-in from floppy does not have a valid DD173 or JANAP-128 of ACP-126 format.	1. Bring file up in Message Preparation Editor. 2. Edit file to conform to format requirements. 3. Save edited file. 4. Reenter command.

It is important that the SAT and AMHS PLA tables be kept in synchronization. If you edit one you must edit the other, should you intend to release messages in any format other than ACP 126. Section 2.5.3 describes the processes for PLA updates in more detail.

The only off-line routine procedures required of the SAT operator consist of maintaining the PLA/RI table and the AIG table used by the SAT software at system startup. To update the PLA/RI table, exit the SAT software and follow the procedures specified below:

(1) Using the PC editor of choice, edit the file **J:\AUTODIN\PLA.R**. Be prepared to specify classified and unclassified RIs for any PLAs being added to the table. It is recommended that this table be kept in alphabetical order by PLA to ease maintenance as the table grows. The format of the table is as follows:

Col 1-7:	Unclassified RI (4-7 characters).
Col 8:	Highest message classification allowed for this station (T,S,C,U).
Col 9-15:	Classified RI (4-7 characters).
Col 16:	Separator (/).
Col 17:	PLA (maximum of 55 characters, with or without blanks).

(2) Once edits are complete, save the edits and exit to the DOS prompt, **J:.**

(3) Enter **BUILDPLA**.

(4) Enter **UPDATE**.

To update the AIG tables, exit the SAT software and follow the procedures specified below:

(1) Using the PC editor of choice, create or edit the file **J:\AUTODIN \AIGNNN** , where the NNN corresponds to the number of the AIG being modified or created. Be prepared to specify the PLAs that comprise the AIG. It is recommended that this table be kept in alphabetical order by PLA to ease maintenance as the table grows.

(2) Once edits are complete, save the edits and exit to the DOS prompt, **J:**.

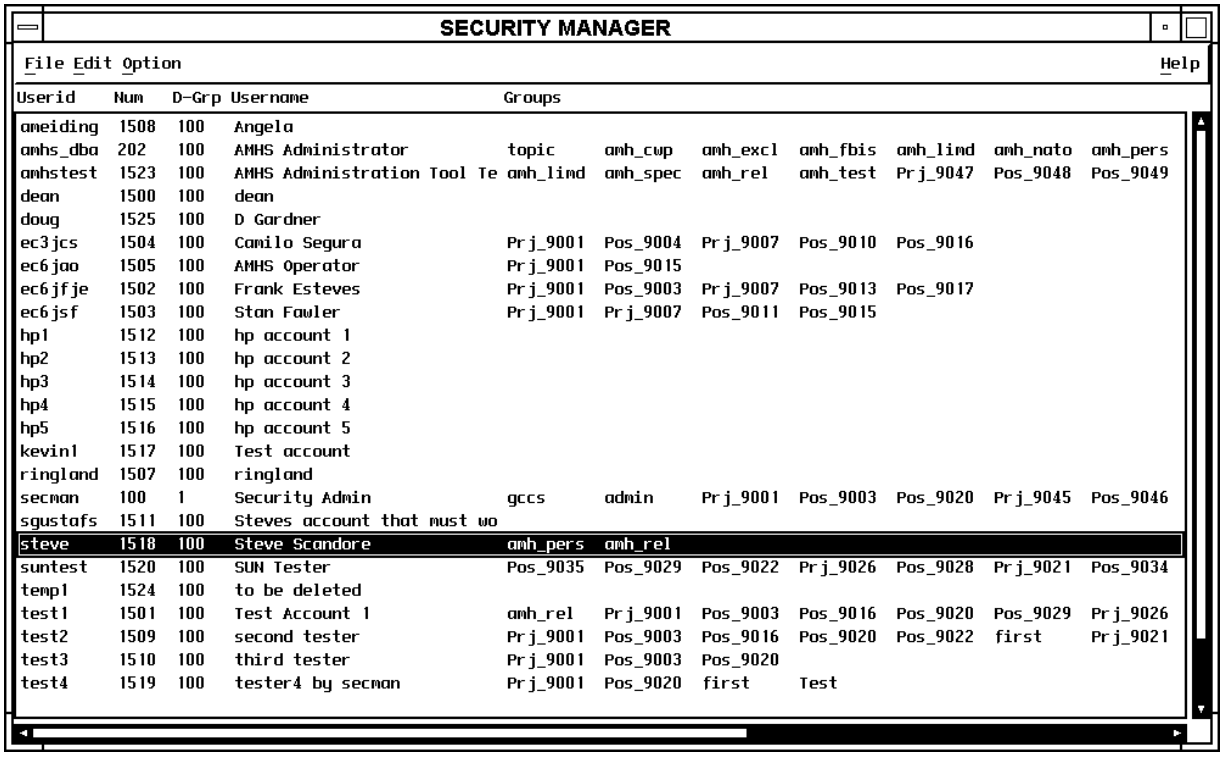
5.16 CREATING ACCOUNTS FOR GCCS VERSION 2.1

GCCS User Account maintenance is normally the responsibility of the GCCS System Administrator but AMHS users must have accounts to access the COE EM Desktop. This section outlines the steps the Sys Admin uses. All users must have a UNIX account called a "user account" as described in the System Administrator's Manual GCCS-SAM-2.1. The following procedures are referenced in Section 9, USER ACCOUNT ADMINISTRATION, of GCCS-SAM-2.1. There are two basic steps in creating a UNIX user account. The first is to create the account. The second is to create the customized profile, if the default profile is not sufficient.

(1) Creating user accounts (performed at the EM Server's console).

(a) Log in as **secman** with proper password.

(b) Select **Prefs** from the menu bar. Select **Change Profile** from the menu. Click the **Next** or **Prev** buttons until **SYSADMIN** is displayed in the **Position:** field. Click the **OK** button.



Userid	Num	D-Grp	Username	Groups						
ameiding	1508	100	Angela							
amhs_dba	202	100	AMHS Administrator	topic	amh_cwp	amh_excl	amh_fbis	amh_limd	amh_nato	amh_pers
amhstest	1523	100	AMHS Administration Tool Te	amh_limd	amh_spec	amh_rel	amh_test	Prj_9047	Pos_9048	Pos_9049
dean	1500	100	dean							
doug	1525	100	D Gardner							
ec3jcs	1504	100	Camilo Segura	Prj_9001	Pos_9004	Prj_9007	Pos_9010	Pos_9016		
ec6jao	1505	100	AMHS Operator	Prj_9001	Pos_9015					
ec6jfje	1502	100	Frank Esteves	Prj_9001	Pos_9003	Prj_9007	Pos_9013	Pos_9017		
ec6jsf	1503	100	Stan Fawler	Prj_9001	Prj_9007	Pos_9011	Pos_9015			
hp1	1512	100	hp account 1							
hp2	1513	100	hp account 2							
hp3	1514	100	hp account 3							
hp4	1515	100	hp account 4							
hp5	1516	100	hp account 5							
kevin1	1517	100	Test account							
ringland	1507	100	ringland							
secman	100	1	Security Admin	gccs	admin	Prj_9001	Pos_9003	Pos_9020	Prj_9045	Pos_9046
sgustafs	1511	100	Steves account that must wo							
steve	1518	100	Steve Scandore	amh_pers	amh_rel					
suntest	1520	100	SUN Tester	Pos_9035	Pos_9029	Pos_9022	Prj_9026	Pos_9028	Prj_9021	Pos_9034
temp1	1524	100	to be deleted							
test1	1501	100	Test Account 1	amh_rel	Prj_9001	Pos_9003	Pos_9016	Pos_9020	Pos_9029	Prj_9026
test2	1509	100	second tester	Prj_9001	Pos_9003	Pos_9016	Pos_9020	Pos_9022	first	Prj_9021
test3	1510	100	third tester	Prj_9001	Pos_9003	Pos_9020				
test4	1519	100	tester4 by secman	Prj_9001	Pos_9020	first	Test			

Figure 5-34. Security Manager Display Window

(c) Double-click the **Security** icon. The **run_security** window comes up. Enter the **secman's** password at the **Password:** prompt. the **Security Manager** window appears.

(d) Select **File** from the menu bar. Select **Create Account** from the menu. The **SECURITY MANAGER: Create Accounts** window appears.

(e) Enter the **USER ID:** (8 characters or less starting with an alphabetic character (a-z) and containing only alphanumeric characters and the underscore (<_>)).

(f) Enter the **USER NAME:** (essentially an administrative comment field—recommend including section and point of contact information such as location and telephone number).

Example: ccj6_doe MAJ John Doe 8-6580

NOTE: Do not use commas or other special characters. Use only letters, numbers and underscore.

(g) The **USER #** field is filled in by the utility. (This is the UID and it is the last used value plus 1. The number may be edited to re-use old UID numbers that have been deleted.)

(h) Enter the **PASSWORD:** (This will be the user's login password.)

(i) Enter the **SYBASE SYS ADMIN USERNAME:** (sa).

(j) Enter the **SYBASE SYS ADMIN PASSWORD:** (as assigned in Section 5.4 of the System Administrator's Manual).

(k) Click the button for the **DEFAULT GROUP:** field. Select from **admin** (for an administrator account) or **gccs** (for a user account). Click the **Apply** button.

(l) Click the button for **OPTIONAL GROUP:** field. Select from **admin** (for an administrator account) or **gccs** (for a user account). Click the **Apply** button.

(m) Click the button for the **Acct_groups** field. Select from **root**, **Security Admin**, **System Admin**, or **GCCS Operator** (for user account). Click the **Apply** button.

(n) Click the button **Role** field. Select from **SSO Default** (user account management and security), **SA Default** (System Administrator, which is primarily used for installing new software segments), or **GCCS Default** (for a user account). Click the **Apply** button.

(o) When all the fields are successfully completed, click the **OK** button on the **SECURITY MANAGER: Create Accounts** window.

(p) Select **File** from the menu bar. Select **Exit** from the menu. Click **OK** to the Exit question.

(q) Reference Figure 5-35.

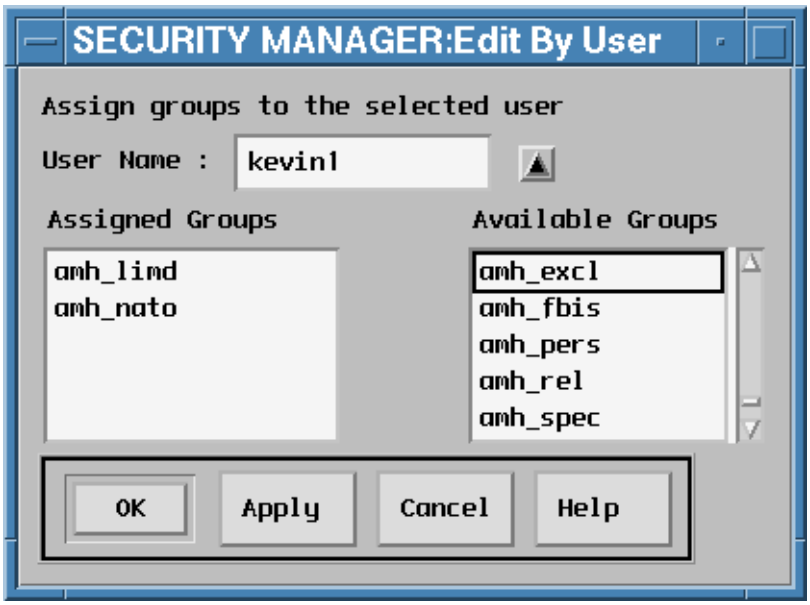


Figure 5-35. Security Manager Edit Window

(2) Customizing profile. After the System Administrator has registered the new user, a default user profile will be created. The following steps are used to customize that user's profile. The following instructions can also be found in Section 9 of the System Administrator's Manual.

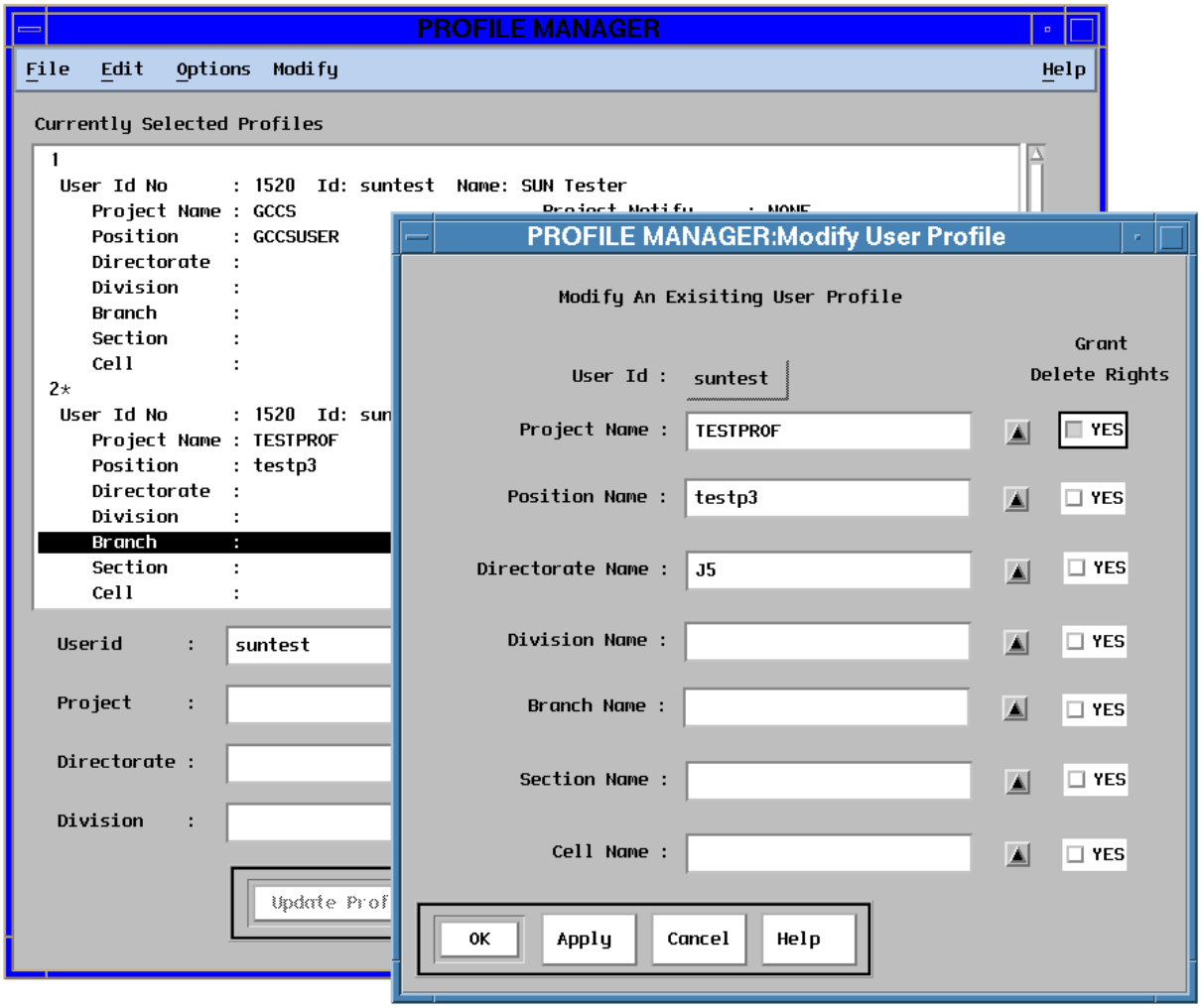


Figure 5-36. Profile Manager Screens

(a) Log in as **secman** with proper password.

(b) Select **Prefs** from the menu bar. Select **Change Profile** from the menu. Click the **Next** or **Prev** buttons until **SYSADMIN** is displayed in the **Position:** field. Click the **OK** button.

(c) Double-click the **Profile** icon. The **Profile Manager** window appears.

(d) Select **File** from the menu bar. Select **Edit New User Profile** from the menu. The **PROFILE MANAGER: Edit New User Profile** window appears.

(e) Click the button for the **User ID:** field. Select appropriate user from the registered users display.

(f) Click the button for the **Project:** field. Select appropriate project from the display.

(g) Click the button for the **Position:** field. Select from **GCCSUSER** or **SYSADMIN** in the position display. (This selection is tied to the user's launch window icon selections.)

NOTE: These first three fields of the window are mandatory for user profile creation. The others deal with the organizational structure for the site. They include Directorate, Division, Branch, Section, and Cell.

(h) Click on the **OK** or **Apply** button. Select **File** from the menu bar and **Exit** from the menu.

At this point, the user's UNIX accounts and user profiles should have been established based on the procedures listed above.

5.17 TROUBLESHOOTING

Error messages on the AMHS Server are routed to the console. The file **/var/adm/messages** contains error messages which have been generated by the Solaris Operating System in addition to other informative messages. When a crontab entry generates an error message, the output is sent as a mail message to the **root** user and can be viewed by invoking the **mail** program. When a line printer error occurs, the error message is written to the **/var/adm/lpd-errs** file.

The error logs described in Section 2.11 will be useful in locating difficulties, especially if they are displayed on the AMHS Administration Status window.

6. REDUNDANT AMHS PROCEDURES

The inherent capability of the GCCS AMHS to run with redundant servers helps fulfill the AUTODIN messaging delivery requirements for mission critical information. As described in previous sections, the AUTODIN system is designed to guarantee delivery of all messages to someone. The AUTODIN store and forward backbone, with SAT message transfer handshaking, completes its responsibility when the SAT acknowledges receipt of each message. At this point the final message delivery is the responsibility of the SAT Operator. The GCCS AMHS is a knowledge based delivery system that assists the SAT Operator (GCCS AMHS Systems Operator) to deliver the messages to the proper addressee. At this point the system reliability depends upon two issues: 1) The reliability of the equipment, and 2) The quality of the procedures followed. See Figure 6-1.

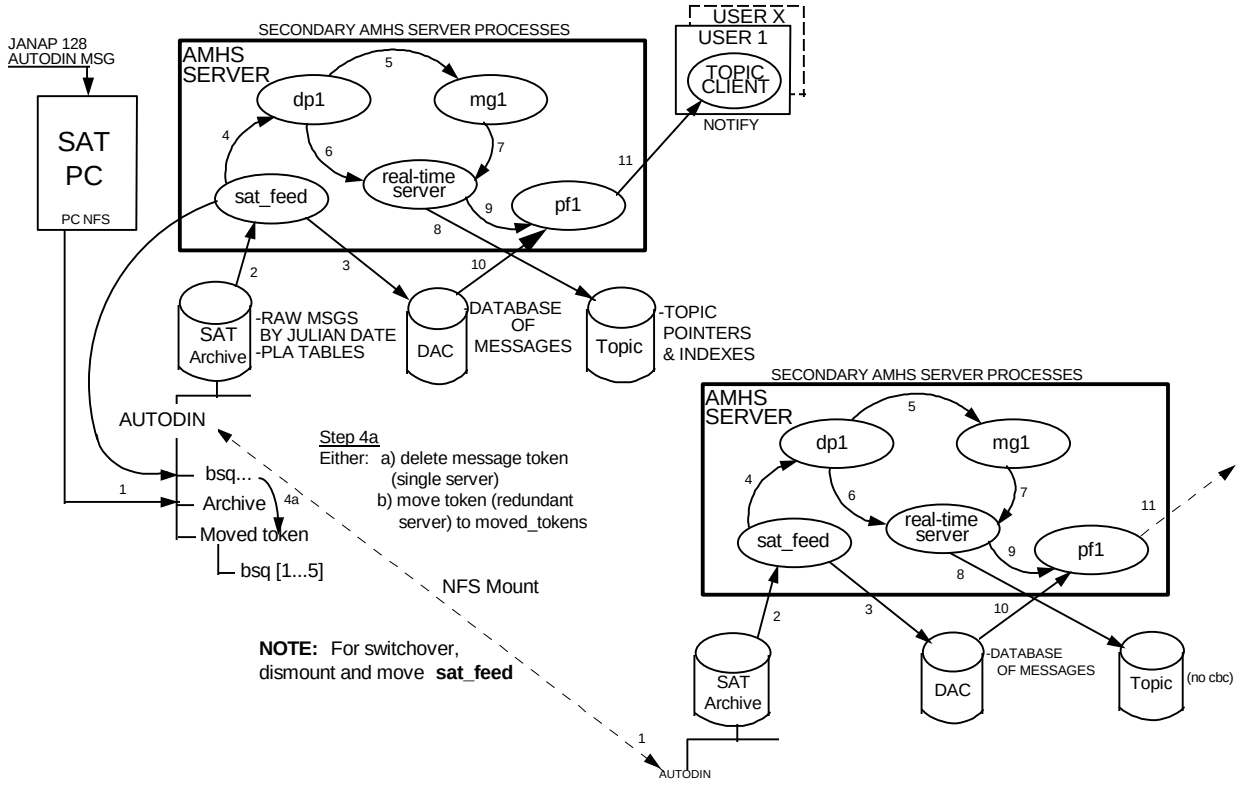


Figure 6-1. Redundant AMHS Functional Block Diagram

A failure in the SAT terminal will not cause any messages to be lost because they are held by the AUTODIN system until the SAT is repaired and back on line. The AMHS Servers are typically Sun SPARC 20s with fault tolerant RAID Level 5 hard drives. A failure in the primary AMHS Server will result in an apparent SAT hard drive error, notifying the AUTODIN system to hold all traffic. All message traffic prior to the AMHS Server failure was being processed on both the

primary and secondary AMHS Servers, so when the secondary server is reconfigured to perform the role of the primary server, no message traffic is lost and the system will continue to function as it did before the fault. Some of the CBCs will be missing, but none of the incoming or outgoing messages are missing. The AMHS depends upon the EM Server for several administrative, monitor, and control functions, but the loss of the EM Server will only affect buckslip routing, message prep, and user logon. The AMHS will continue to process incoming messages. If the EM Server is properly restored no system degradation will be seen.

The second AMHS Server in the redundant configuration is only processing incoming messages and not responding to user retrospective searches. These functions are being handled by the primary AMHS Server. This decreases the CPU loading so this machine could easily double as the AMHS System / Database Administrators system and could be used for these other tasks. This would significantly reduce the total cost of the redundant system.

6.1 INSTALLATION AND CONFIGURATION

This section describes the procedures to install the AMHS in a two-server configuration with one server acting as a primary and the other acting as a secondary. The second part includes some guideline information on switching between servers when one fails.

For purposes of these procedures, the primary server will have a hostname of ‘nmccamh1’ and the secondary server will have a hostname of ‘nmccdb2’. Of course, you should substitute the hostnames for your particular site.

The first step is to install and test each AMHS server independently. Install the first server as instructed during the AMHS installation course. After the first server has been successfully installed and tested, start on the second server’s installation. Independently verify the second server’s installation by running the same tests performed on the first server. Remember to update the **c:\nfs\drives.bat** file on the SAT PC to point to the second server (reboot the SAT PC after making the change). Note: It is not necessary to update the ‘amhserver’ hostname alias to test the second server’s installation. The alias only needs to be change to test client workstations.

After testing both servers and verifying that both work independently as AMHS servers, the steps to configure them in a primary and secondary configuration can be performed. Do not perform the steps until both servers have been tested to work independently of each other. Note: Restore the c:\nfs\drives.bat file on the SAT PC before proceeding. Also, restore the ‘amhserver’ alias if changed to test client workstations.

The **c:\nfs\drives.bat** file on the SAT PC should always point to the primary AMHS server (in this case ‘nmccamh1’).

6.2 SPECIFIC PROCEDURES FOR BOTH SERVERS

Execute the following on both ‘nmccamh1’ and ‘nmccdb2’:

(1) Log in as the **amhs_dba** .

(2) Create the following directories:

```
mkdir /amhs/sat/autodin/moved_tokens
mkdir /amhs/sat/autodin/moved_tokens/bsq1
mkdir /amhs/sat/autodin/moved_tokens/bsq2
mkdir /amhs/sat/autodin/moved_tokens/bsq3
mkdir /amhs/sat/autodin/moved_tokens/bsq4
mkdir /amhs/sat/autodin/moved_tokens/bsq5
```

(3) Create the following soft link:

ln -s /amhs/sat/autodin/archive /amhs/sat/autodin/moved_tokens/archive .

6.3 SPECIFIC PROCEDURES FOR PRIMARY SERVER (nmccamh1)

Execute the following procedures on the primary server (nmccamh1):

(1) Log in as the **amhs_dba** .

(2) Stop the AMHS server processes.

(3) Edit the vardef file:

vi /h/AMHS/Server/topic/amhs_db/vardef .

(4) This file contains a commented entry for a ‘move_token’ variable. Uncomment the entry and change the value:

move_token=/h/AMHS/Server/sat/autodin/moved_tokens/bsq3 .

6.4 SPECIFIC PROCEDURES FOR SECONDARY SERVER (nmccdb2)

Execute the following procedures on the secondary server (nmccdb2):

(1) Log in as the **amhs_dba** .

(2) Stop the AMHS server processes.

(3) Edit the vardef file:

vi /h/AMHS/Server/topic/amhs_db/vardef .

(4) This file contains an entry for a **cfe_dir** variable. Change to the following:

cfe_dir=/h/AMHS/Server/sat/autodin/moved_tokens .

(5) Edit the **vfstab** file:

vi /etc/vfstab .

(6) Add the following line to the vfstab file:

nmccamh1:/amhs/sat/autodin - /amhs/sat/autodin nfs - yes rw,bg,soft .

(7) Reboot the workstation.

6.5 TESTING THE INSTALLATION

Testing the dual server configuration is very simple. Remember that the SAT terminal is the source of all incoming AMHS messages. Once a message is received at the SAT terminal, the primary server (nmccamh1) will process the message into its own TOPIC database. Once the message has been processed by the primary server (nmccamh1), the message will be passed to the secondary server (nmccdb2). The secondary server (nmccdb2) will process the message into its own TOPIC database. In other words, two separate TOPIC databases will be maintained on two separate servers as each message is received.

To test, start all the AMHS processes on both servers. Test the configuration by sending a test message into the system. This can be done by releasing a message from any client workstation or by sending a message from the SAT PC in loop back. Verify the message was received and processed by both AMHS servers by checking the AMHS message browsers from both the primary and the secondary servers. An AMHS message browser launched and displayed on the primary server will display messages from the primary TOPIC database. Likewise, an AMHS message browser launched and displayed on the secondary server will display messages that are on the secondary server's TOPIC database. The message should appear in both databases. Note that comeback copies (CBCs) will not appear in the secondary server's TOPIC database.

The SAT/CBT terminal is the source of all incoming AMHS messages. Once a message is received at the SAT terminal, the primary server (nmccamh1) will process the message into its own TOPIC database. Once the message has been processed by the primary server (nmccamh1), the message will be passed to the secondary server (nmccdb2). The secondary server (nmccdb2) will process the message into its own TOPIC database. In other words, two separate TOPIC databases will be maintained on two separate servers as each message is received. To test, start all the AMHS processes on both servers. Test the configuration by sending a test message into the system. This can be done by releasing a message from any client workstation or by sending a message from the SAT PC in loop back. Verify the message was received and processed by both AMHS servers by checking the AMHS message browsers from both the primary and the secondary servers. An AMHS message browser launched and displayed on the primary server will display messages from the primary TOPIC database. Likewise, an AMHS message browser launched and displayed on the secondary server will display messages that are on the secondary server's TOPIC database. The message should appear in both databases. Note: Come-back-copies (CBCs) will not appear in the secondary server's TOPIC database.

6.6 HARDWARE CONFIGURATION

Two redundant AMHS threads are available. Each receives AUTODIN (AMME) traffic. Each thread consists of a pair of CPUs. Both threads connect to the Local Area Network (LAN) The operational thread is called the "PRIMARY THREAD", while the backup thread is called the "SECONDARY THREAD". Either CPU can be the PRIMARY THREAD at any given time. For purposes of this discussion, the first CPU, **amh_sv1** has been designated as the PRIMARY THREAD.

NOTE: The PRIMARY THREAD and SECONDARY THREAD designations will change with each switchover performed.

6.7 MODES OF OPERATION

In the Normal mode, both the PRIMARY and SECONDARY THREADS are up and on the LAN. The SECONDARY THREAD is NFS-mounted to PRIMARY THREAD for access to the SAT subdirectories and the SAT PC is NFS-mounted to PRIMARY THREAD and cabled to AMME or appropriate AUTODIN interface.

In the Degraded mode, one thread is down for Repair (bad CPU or bad disk) or software testing or the SAT PC is down. When the AMHS is in the degraded mode the PRIMARY THREAD must be closely watched and the SAT placed off line if any problems appear to stop message traffic.

6.8 SOFTWARE AND DATA FILE SYNCHRONIZATION

Data and software on the two AMHS threads are kept synchronized by a script that is run on the PRIMARY THREAD once a day by the crontab daemon. The amhs_dba makes changes on the PRIMARY THREAD (updating user profiles, system topics, adding users, etc.), and these changes are copied to the SECONDARY THREAD each day when the root crontab entry is invoked. The script that performs the synchronization may be run manually, by "root", to ensure that essential data and software on the two threads are identical.

/h/AMHS/Server/Scripts/admin/sync_amhs

6.9 WHEN TO INITIATE SWITCHOVER

Under the normal mode of operation, a hardware failure of the PRIMARY THREAD is the only reason for initiating a switchover. If the SECONDARY THREAD experiences a failure, the AMHS is considered to be in a degraded mode of operation and the failure must be corrected as soon as possible. Hardware failures that can lead to switchover are categorized as either a thread failure or a SAT PC failure.

Before switchover is initiated, the System Administrator(s) must determine the following:

(1) Extent/location of failure (which hardware or software components have failed).

(2) Estimated time to repair (total downtime anticipated).

NOTE: If the estimated time to repair is longer than the currently acceptable down time of approximately 30 minutes, then switchover should be initiated.

A switchover may also be performed in conjunction with the monthly level 0 backup, minimizing downtime to AMHS users and keeping the switchover procedures current. The steps to accomplish this are:

(1) Perform level 0 backup of the SECONDARY THREAD.

(2) Shut down both threads.

(3) Perform the switchover bringing up the alternate pair of CPUs as the PRIMARY THREAD.

(4) Backup the second set of CPUs now running as the SECONDARY THREAD.

(5) Bring the secondary thread back on line.

(6) Update the System Monitor on the Sun EM Server to reflect the new configuration. This is done by running the **active_spt** script.

6.9.1 Preliminary Switchover Procedures

A catastrophic failure of the AMHS PRIMARY THREAD will usually preclude running any steps prior to the switchover. When a switchover can be scheduled, however, the following steps should be performed before starting the switchover procedures:

(1) Log in to the PRIMARY THREAD as **amhs_dba** and run the "SYNCHRONIZE AMHS" procedure to ensure that essential data and software on the two threads are identical.

NOTE: This procedure is scheduled to automatically run out of crontab once each day.

/usr/topic/amhs_db/tool/sync_amhs

(2) Log in to the SECONDARY THREAD Server as the **amhs_dba** and run the script **clear_user_queues** to clear the user queues on the SECONDARY THREAD.

/h/AMHS/Server/Scripts/admin/Clean_Queues

(3) To ensure that users receive all their messages, enable profiling on the SECONDARY THREAD for some predetermined period of time before shutting down profiling on the PRIMARY THREAD (a half-hour overlap of profiling on the two threads is recommended).

(a) Type **cd /h/AMHS/Server/topic/amhs_db;rm pf?.pfx .**

(b) Type **topic_cmd** as the **amhs_dba** on the SECONDARY THREAD.

(c) Enter **option 3** to select profiling startup.

(d) Enter **pf.** to activate all profile processes.

6.9.2 Detailed Switchover Procedures

Since the complete switchover will usually take longer than 30 minutes, notify the AUTODIN Switching Center point of contact to reroute high-precedence (flash and flash override) traffic to some other delivery method if necessary. Otherwise, hold the traffic.

(1) Launch the GCCS System Monitor's pull-down system alarms and create a message to notify users that the AMHS will be inaccessible until further notice. Users should also be requested to log off the AMHS until the switchover has been completed. (See Appendix C, Section C.3.5.)

(2) Shut down the SAT PC software with the following sequence:

(a) Highlight **INITIALIZATION FUNCTIONS** option then press **<ENTER>** .

(b) Highlight **EXIT TO DOS** then press **<ENTER>** .

(c) Enter **Y** to Yes/No then press **<ENTER>** .

(3) Power down SAT PC1.

(4) Log into the PRIMARY Server as the **amhs_dba** and use **topic_cmd** to shut down all AMHS processes:

(a) Enter option **0** to see the status of the processes.

(b) Enter option **31** to shut down TOPIC processes.

(c) Verify that no processes are running.

(d) Enter **.** to exit **topic_cmd** .

If the printed message indicates that a process is still active, the process should be manually killed from the command line as follows:

(e) For TOPIC processes (**pf1**, **mg1**, **dp1**, **dp6**, server, etc.):

1) Type **topicd** .
2) Enter **rtsend mailbox <process name> EXIT**, where process name is the TOPIC process - - **pf1**, **mg1**, **dp1**, **dp6**, etc.
3) Repeat the command for each TOPIC process running.

(f) For non-TOPIC processes (**monitor_q**, **reuter_copy**):

1) Type **ps -aux | grep <process name>** .
2) Type **kill <pid>** (PID = process ID number).

(5) Log in to the SECONDARY Server as the **amhs_dba** and shut down all AMHS processes using the same steps just performed on the PRIMARY Server.

(6) On the SECONDARY Server, remove the NFS mount to the PRIMARY Server by performing the following steps:

(a) Gain root permissions by typing
su and enter the root password at the prompt.

(b) Check that a mount exists to the SECONDARY Server for /amhs/sat/autodin by typing: **df -k** .

(c) Unmount this directory by typing:
umount /amhs/sat/autodin.

(d) Check that the mount to the SECONDARY Server is gone by typing **df -k** again.

(7) Edit the **/etc/vfstab** file on the SECONDARY Server to comment out the NFS mount for **/amhs/sat/autodin** .

#nmccamh2:/amhs/sat/autodin - /amhs/sat/autodin nfs - yes rw,bg,soft

(8) Power up the **SAT PC1** and Ctrl+C out of the boot up sequence to get to the "**C:**" prompt. Reconfigure the SAT to point to the SECONDARY Server with the following sequence:

(a) From the C: drive, type **cd nfs** then press **<ENTER>** .

(b) Edit the **drivers.bat** file.

(c) Change the hostname of the mount to point to the SECONDARY Server.

(d) The SAT software should now be brought up with the reboot.

NOTE: **DO NOT PUT THE SAT IN THE ON-LINE MODE AT THIS POINT!!**

The PRIMARY Server is the <u>other</u> AMHS hardware configuration (previously the SECONDARY Server), and vice versa. The remaining steps are written with this assumption.

(9) Log in to the new PRIMARY Server as the **amhs_dba** and start up all AMHS processes using **topic_cmd**:

Option 1 (Startup > Topic)

(10) Verify that all essential processes are running.

(11) Reset all workstations to point to the new PRIMARY Server by logging into the EM Server and updating the **amhserver** hostname alias.

(a) Log into the EM Server as **root** and edit the hosts file.

```
cd /h/EM/nis_files
vi hosts
```

(b) Change the location of the **amhserver** alias from the old server to the new server.

(c) Recompile the host file by running the update and **make_hosts** scripts:

(d) It is necessary for the Users to reboot their workstations to be operating on the new PRIMARY Server.

(12) Verify that the new PRIMARY Server is accessible and operating correctly by invoking the AMHS from the operator's workstation. (Reboot it first .)

NOTE: The SAT creates a file for each newly arrived message by naming it with the DTG (e.g. DTG=271400Z JUL 93 creates a file named 271400). Before creating the filename, however, the SAT first checks the archive subdirectory and, if a file already exists by that name, the SAT creates a unique filename by adding the Station Serial Number (SSN) as a filename extension.

When a switchover is performed, the SAT PC begins creating a new archive subdirectory, /usr/amhs/sat/autodin/archive/r<julian day>, for inbound AUTODIN messages on the newly designated PRIMARY SERVER. This new archive subdirectory does not contain any of the messages that were present on the other thread. If the SAT now receives a message with a DTG that had previously been received, it will place the message file in the archive without a filename extension. The sat_feed will then copy this file over to the appropriate /usr/dac subdirectory and OVERWRITE THE PREVIOUSLY LOADED MESSAGE FILE. The AMHS database will now contain two different entries, each with a different set of Classification, Precedence, From, and Subject, but with a pointer to the most recently received message.

To prevent this from occurring, the message files in the /usr/amhs/sat/autodin/archive/r<julian day> subdirectory on what is now the SECONDARY THREAD (the thread that had been directly receiving AUTODIN messages from the SAT) must be copied over to the new PRIMARY THREAD. The SAT will then properly create unique file names.

This copy is performed with the following procedures:

(c) Log in to the new PRIMARY THREAD Server as **amhs_dba** .

(d) Type **cd /usr/amhs/sat/autodin/archive/** .

(e) Type **mkdir r<julian day>** if it doesn't already exist.

(f) Type **chown amhs_dba r<julian day>** .

(g) Type **chgrp gccs r<julian day>** .

(h) Type **cd r<julian day>** .

(i) Type **ftp (old server name)** .

amhs_dba
<password for amhs_dba>
cd /usr/amhs/sat/autodin/archive/r<julian_day>
binary
prompt n
mget *
quit

(13) Put the SAT PC in on-line mode and verify that messages are being received by the new PRIMARY SERVER

(14) Call the AUTODIN Switching Center point of contact and ask them to return high precedence message routing to AMHS.

(15) Generate a system alarm to notify users that the AMHS is operational and they may now reboot and log back in.

(16) Repair the SECONDARY THREAD and bring it back on-line.

6.9.3 SAT PC Failure Recovery Process

When a SAT fails, the recovery process depends upon the site-specific plan. If your site has a hot spare, you need only move the input communication connection to the backup SAT, reboot it and restart the SAT. At a minimum, the site should have a spare CCPII card, duplicate LAN card, and bootable PC hard drive preloaded with preconfigured software. With this, most any X86 ISA machine can serve as the temporary SAT.

The SAT software automatically shuts down and stops receiving AUTODIN traffic when any failure is detected. When the SAT shuts down, the AMME queues up (holds) messages for the AMHS, so no message traffic is lost. If a failure appears to be due to the PC (as opposed to communications hardware or a problem on the AMHS Server), the following steps should be used to reconnect and configure the spare SAT to replace the failed operational SAT PC. This procedure should take 5-10 minutes. Users will continue to have access to the AMHS while the swapout is in progress.

NOTE: If the personnel performing this procedure are unfamiliar with this switchover, or it appears that the failure may be caused by more than just the SAT PC, the AUTODIN Switching Center should be notified to reroute high-precedence (flash) traffic.

(1) Using the system monitor, generate a system alarm to notify users that the AMHS will not be receiving live AUTODIN messages for approximately N minutes.

(2) Shut down the SAT PC software with the following commands:

(a) Highlight **INITIALIZATION FUNCTIONS** option then press **<ENTER>** .

(b) Highlight **EXIT TO DOS** then press **<ENTER>** .

(c) Enter **Y** to Yes/No then press **<ENTER>** .

(3) Power down SAT PC1.

(4) Remove AMME cable from SAT PC1 and connect to backup SAT PC2.

(5) Power up the SAT PC2 .

NOTE: If you can move the LAN card from the SAT PC1 to SAT PC2 and your hard drive is a preconfigured mirror of SAT PC1 skip to step (d).

(a) Under the Network column, highlight the **Direct Connect** option then press **<ENTER>** .

(b) Verify/select the following on three (3) different screens.

NOTE: An **<ENTER>** on the last item moves to the next screen. TABS and BACKTABS should be used to move between items on a screen.

1) Screen 1 of 2

Time:	GMTO
Daylight Savings Time?	N
Network information service?	N
RARP?	N

2) Screen 2 of 2

PC Name:	amh_sat
IP address...:	xxx.xxx.xxx.x
PCNFS server name:	amh_sv(1 or 2)
IP address of PCNFS servr:	xxx.xxx.xxx.x
Advanced Questions?	Y

3) Advanced Questions

User name:	amhs_dba
Gateway name:	xxxxxx(site-specific)
IP address of gateway:	xxx.xxx.xxx.x (appears when highlighted)
Subnet mask:	255.255.255.0

(c) A message confirming that changes are made appears. Enter **Y** to save the configuration and reboot.

(d) The SAT software should now come up with reboot.

(6) Put SAT PC2 in on-line mode and verify that messages are being received by PRIMARY THREAD.

(7) Call AMME POOCH and request they return high-precedence routing to AMHS.

(8) Generate system alarm to notify users that AMHS is operational again.

This page intentionally left blank.